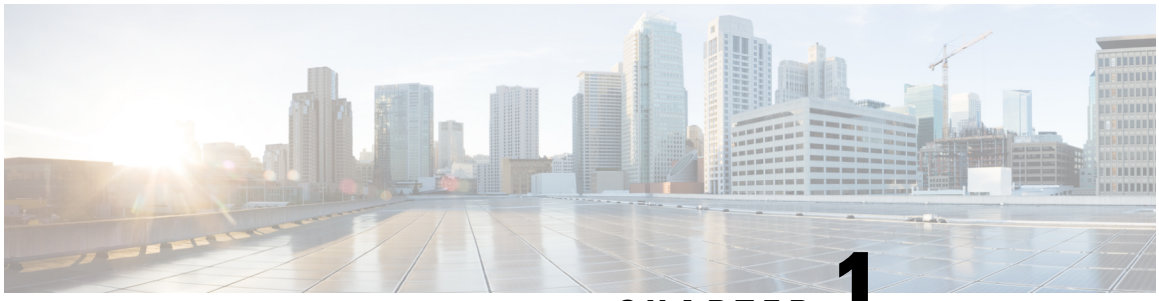
Cisco Firepower 4100/9300 FXOS CLI Configuration Guide, 1.1(4)

First Published: 2016-03-20

Last Modified: 2020-07-02

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

CONTENTS



CHAPTER 1

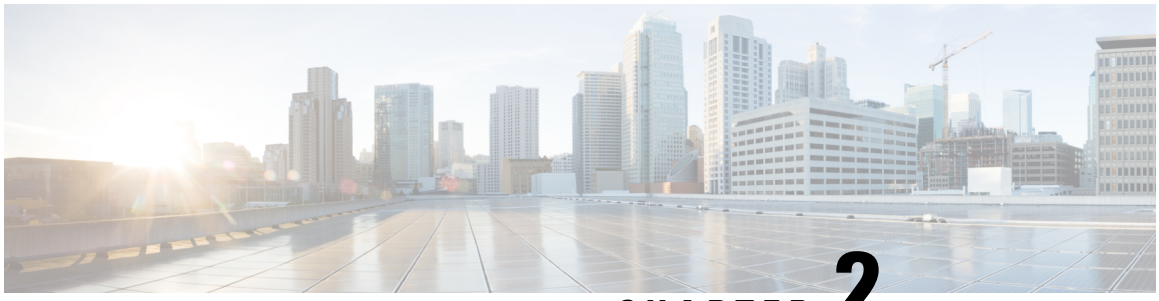
Introduction to the Firepower Security Appliance

About the Firepower Security Appliance

The Cisco Firepower 4100/9300 chassis is a next-generation platform for network and content security solutions. The Firepower 4100/9300 chassis is part of the Cisco Application Centric Infrastructure (ACI) Security Solution and provides an agile, open, secure platform that is built for scalability, consistent control, and simplified management.

The Firepower 4100/9300 chassis provides the following features:

- Modular chassis-based security system—provides high performance, flexible input/output configurations, and scalability.
- Firepower Chassis Manager—graphical user interface provides streamlined, visual representation of current chassis status and simplified configuration of chassis features.
- FXOS CLI—provides command-based interface for configuring features, monitoring chassis status, and accessing advanced troubleshooting features.
- FXOS REST API—allows users to programmatically configure and manage their chassis.



CHAPTER 2

CLI Overview

Managed Objects

The Firepower eXtensible Operating System (FXOS) uses a managed object model, where managed objects are abstract representations of physical or logical entities that can be managed. For example, chassis, security modules, network modules, ports, and processors are physical entities represented as managed objects, and licenses, user roles, and platform policies are logical entities represented as managed objects.

Managed objects may have one or more associated properties that can be configured.

Command Modes

The CLI is organized into a hierarchy of command modes, with EXEC mode being the highest-level mode of the hierarchy. Higher-level modes branch into lower-level modes. You use **create**, **enter**, and **scope** commands to move from higher-level modes to modes in the next lower level, and you use the **up** command to move up one level in the mode hierarchy. You can also use the **top** command to move to the top level in the mode hierarchy.

Note Most command modes are associated with managed objects, so you must create an object before you can access the mode associated with that object. You use **create** and **enter** commands to create managed objects for the modes being accessed. The **scope** commands do not create managed objects and can only access modes for which managed objects already exist.

Each mode contains a set of commands that can be entered in that mode. Most of the commands available in each mode pertain to the associated managed object.

The CLI prompt for each mode shows the full path down the mode hierarchy to the current mode. This helps you to determine where you are in the command mode hierarchy, and it can be an invaluable tool when you need to navigate through the hierarchy.

The following table lists the main command modes, the commands used to access each mode, and the CLI prompt associated with each mode.

Table 1: Main Command Modes and Prompts

Mode Name	Commands Used to Access	Mode Prompt
EXEC	**top** command from any mode	`#`
Adapter	**scope adapter** command from EXEC mode	`/adapter #`
Cabling	**scope cabling** command from EXEC mode	`/cabling #`
Chassis	**scope chassis** command from EXEC mode	`/chassis #`
Ethernet server domain	**scope eth-server** command from EXEC mode; this command and all subcommands are currently not supported	`/eth-server #`
Ethernet uplink	**scope eth-uplink** command from EXEC mode	`/eth-uplink #`
Fabric interconnect	**scope fabric-interconnect** command from EXEC mode	`/fabric-interconnect #`
Firmware	**scope firmware** command from EXEC mode	`/firmware #`
Host Ethernet interface	**scope host-eth-if** command from EXEC mode **Note** This command and all subcommands are not supported at this level; the Host Ethernet interface commands are available in `/adapter #` mode.	`/host-eth-if #`
License	**scope license** command from EXEC mode	`/license #`
Monitoring	**scope monitoring** command from EXEC mode	`/monitoring #`
Organization	**scope org** command from EXEC mode	`/org #`
Packet capture	**scope packet-capture** command from EXEC mode	`/packet-capture #`
Security	**scope security** command from EXEC mode	`/security #`

Mode Name	Commands Used to Access	Mode Prompt
Server	**scope server** command from EXEC mode	`/server #`
Service profile	**scope service-profile** command from EXEC mode **Note** Do not alter or configure service profiles; that is, do not use the **create**, **set**, or **delete** subcommand sets.	`/service-profile #`
SSA	**scope ssa** command from EXEC mode	`/ssa #`
System	**scope system** command from EXEC mode	`/system #`
Virtual HBA	**scope vhba** command from EXEC mode **Note** This command and all subcommands are currently not supported.	`/vhba #`
Virtual NIC	**scope vnic** command from EXEC mode	`/vnic #`

Object Commands

Four general commands are available for object management:

- **create** *object*
- **delete** *object*
- **enter** *object*
- **scope** *object*

You can use the **scope** command with any managed object, whether a permanent object or a user-instantiated object. The other commands allow you to create and manage user-instantiated objects. For every **create** *object* command, a corresponding **delete** *object* and **enter** *object* command exists.

In the management of user-instantiated objects, the behavior of these commands depends on whether the object exists, as described in the following tables:

Table 2: Command Behavior If The Object Does Not Exist

Command	Behavior
create *object*	The object is created and its configuration mode, if applicable, is entered.
delete *object*	An error message is generated.
enter *object*	The object is created and its configuration mode, if applicable, is entered.
scope *object*	An error message is generated.

Table 3: Command Behavior If The Object Exists

Command	Behavior
create *object*	An error message is generated.
delete *object*	The object is deleted.
enter *object*	The configuration mode, if applicable, of the object is entered.
scope *object*	The configuration mode of the object is entered.

Complete a Command

You can use the **Tab** key in any mode to complete a command. Partially typing a command name and pressing **Tab** causes the command to be displayed in full or to the point where you must enter another keyword or an argument value.

Command History

The CLI stores all commands used in the current session. You can step through the previously used commands by using the up-arrow or down-arrow keys. The up-arrow key moves to the previous command in the history, and the down-arrow key moves to the next command in the history. When you get to the end of the history, pressing the down-arrow key does nothing.

You can enter any command in the history again by stepping through the history to recall that command and then pressing **Enter**. The command is entered as if you had manually typed it. You can also recall a command and change it before you press **Enter**.

Commit, Discard, and View Pending Commands

When you enter a configuration command in the CLI, the command is not applied until you enter the **commit-buffer** command. Until committed, a configuration command is pending and can be discarded by entering a **discard-buffer** command.

You can accumulate pending changes in multiple command modes and apply them together with a single **commit-buffer** command. You can view the pending commands by entering the **show configuration pending** command in any command mode.

Note All pending commands are checked for validity. However, if any queued command fails during commit, the remaining commands are applied; failed commands are reported in an error message.

While any commands are pending, an asterisk (*) appears before the command prompt. The asterisk disappears when you enter the **commit-buffer** command.

The following example shows how the prompts change during the command entry process:

```
Firepower# scope system
Firepower /system # scope services
Firepower /system/services # create ntp-server 192.168.200.101
Firepower /system/services* # show configuration pending
 scope services
+    create ntp-server 192.168.200.101
 exit
Firepower /system/services* # commit-buffer
Firepower /system/services #
```

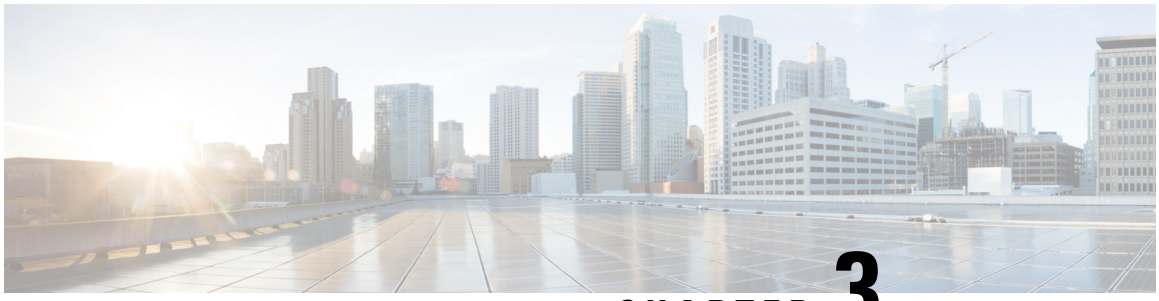
Inline Help for the CLI

At any time, you can enter the **?** character to display the options available at the current state of the command syntax.

If you have not entered anything at the prompt, entering **?** lists all available commands for the mode you are in. With a partially entered command, entering **?** lists all keywords and arguments available at your current position in the command syntax.

CLI Session Limits

FXOS limits the number of CLI sessions that can be active at one time to 32 total sessions. This value is not configurable.



CHAPTER 3

Getting Started

Task Flow

The following procedure shows the basic tasks that should be completed when configuring your Firepower 4100/9300 chassis.

Procedure

Step 1	Configure the Firepower 4100/9300 chassis hardware (see the *Cisco Firepower Security Appliance Hardware Installation Guide*).
Step 2	Complete the initial configuration (see Initial Configuration Using Console Port, on page 9).
Step 3	Set the Date and Time (see Setting the Date and Time, on page 57).
Step 4	Configure a DNS server (see Configuring DNS Servers, on page 102).
Step 5	Register your product license (see License Management for the ASA, on page 15).
Step 6	Configure users (see User Management, on page 27).
Step 7	Perform software updates as required (see Image Management, on page 45).
Step 8	Configure additional platform settings (see Platform Settings, on page 55).
Step 9	Configure interfaces (see Interface Management, on page 105).
Step 10	Create logical devices (see Logical Devices, on page 115).

Initial Configuration Using Console Port

Before you can use Firepower Chassis Manager or the FXOS CLI to configure and manage your system, you must perform some initial configuration tasks using the FXOS CLI accessed through the console port.Use the following procedure to perform initial configuration using the FXOS CLI accessed through the console port.

The first time that you access the Firepower 4100/9300 chassis using the FXOS CLI, you will encounter a setup wizard that you can use to configure the system.

Note To repeat the initial setup, you need to erase any existing configuration using the following commands:

```
Firepower-chassis# connect local-mgmt
firepower-chassis(local-mgmt)# erase configuration
```

You can choose to either restore the system configuration from an existing backup file, or manually set up the system by going through the Setup wizard. If you choose to restore the system, the backup file must be reachable from the management network.

You must specify only one IPv4 address, gateway, and subnet mask, or only one IPv6 address, gateway, and network prefix for the single management port on the Firepower 4100/9300 chassis. You can configure either an IPv4 or an IPv6 address for the management port IP address.

Before you begin

1. Verify the following physical connections on the Firepower 4100/9300 chassis:
   - The console port is physically connected to a computer terminal or console server.
   - The 1 Gbps Ethernet management port is connected to an external hub, switch, or router.

   For more information, refer to the *Cisco Firepower Security Appliance Hardware Installation Guide*.
2. Verify that the console port parameters on the computer terminal (or console server) attached to the console port are as follows:
   - 9600 baud
   - 8 data bits
   - No parity
   - 1 stop bit
3. Gather the following information for use with the setup script:
   - New admin password
   - Management IP address and subnet mask
   - Gateway IP address
   - Hostname and domain name
   - DNS server IP address

Procedure

Step 1 Power on the chassis.

Step 2 Connect to the serial console port using a terminal emulator.

The Firepower includes an RS-232–to–RJ-45 serial console cable. You might need to use a third party serial-to-USB cable to make the connection. Use the following serial parameters:

- 9600 baud
- 8 data bits
- No parity
- 1 stop bit

Step 3 Complete the system configuration as prompted.

Example:

```
Enter the setup mode; setup newly or restore from backup. (setup/restore) ? setup
You have chosen to setup a new Fabric interconnect. Continue? (y/n): y
Enforce strong password? (yes/no) [y]: n
Enter the password for "admin": Farscape&32
Confirm the password for "admin": Farscape&32
Enter the system name:  firepower-9300
Physical Switch Mgmt0 IP address : 10.80.6.12
Physical Switch Mgmt0 IPv4 netmask : 255.255.255.0
IPv4 address of the default gateway : 10.80.6.1
Configure the DNS Server IP address? (yes/no) [n]: y
  DNS IP address : 10.164.47.13
Configure the default domain name? (yes/no) [n]: y
  Default domain name : cisco.com

Following configurations will be applied:
  Switch Fabric=A
  System Name=firepower-9300
  Enforced Strong Password=no
  Physical Switch Mgmt0 IP Address=10.89.5.14
  Physical Switch Mgmt0 IP Netmask=255.255.255.192
  Default Gateway=10.89.5.1
  IPv6 value=0
  DNS Server=72.163.47.11
  Domain Name=cisco.com

Apply and save the configuration (select 'no' if you want to re-enter)? (yes/no): y
Applying configuration. Please wait...  Configuration file - Ok
.....

Cisco FPR Series Security Appliance
firepower-9300 login:  admin
Password: Farscape&32
Successful login attempts for user 'admin' : 1
Cisco Firepower Extensible Operating System (FX-OS) Software
TAC support: http://www.cisco.com/tac
Copyright (c) 2009-2019, Cisco Systems, Inc. All rights reserved.

[...]

firepower-chassis#
```

Accessing the FXOS CLI

You can connect to the FXOS CLI using a terminal plugged into the console port. Verify that the console port parameters on the computer terminal (or console server) attached to the console port are as follows:

- 9600 baud
- 8 data bits
- No parity
- 1 stop bit

You can also connect to the FXOS CLI using SSH and Telnet. The Firepower eXtensible Operating System supports up to eight simultaneous SSH connections. To connect with SSH, you need to know the hostname or IP address of the Firepower 4100/9300 chassis.

Use one of the following syntax examples to log in with SSH, Telnet, or Putty:

Note SSH log in is case-sensitive.

From a Linux terminal using SSH:

- **ssh ucs-***auth-domain**username*@{*UCSM-ip-address* | *UCMS-ipv6-address*}

```
ssh ucs-example\\jsmith@192.0.20.11
ssh ucs-example\\jsmith@2001::1
```

- **ssh -l ucs-***auth-domain**username* {*UCSM-ip-address*| *UCSM-ipv6-address*| *UCSM-host-name*}

```
ssh -l ucs-example\\jsmith 192.0.20.11
ssh -l ucs-example\\jsmith 2001::1
```

- **ssh** {*UCSM-ip-address* | *UCSM-ipv6-address* | *UCSM-host-name*} **-l ucs-***auth-domain**username*

```
ssh 192.0.20.11 -l ucs-example\\jsmith
ssh 2001::1 -l ucs-example\\jsmith
```

- **ssh ucs-***auth-domain**username*@{*UCSM-ip-address* | *UCSM-ipv6-address*}

```
ssh ucs-ldap23\\jsmith@192.0.20.11
ssh ucs-ldap23\\jsmith@2001::1
```

From a Linux terminal using Telnet:

Note Telnet is disabled by default. See Configuring Telnet, on page 63 for instructions on enabling Telnet.

- **telnet ucs-***UCSM-host-name* **ucs-***auth-domain**username*

```
telnet ucs-qa-10
login: ucs-ldap23\blradmin
```

- **telnet ucs-**{*UCSM-ip-address* | *UCSM-ipv6-address*}*ucs-auth-domain**username*

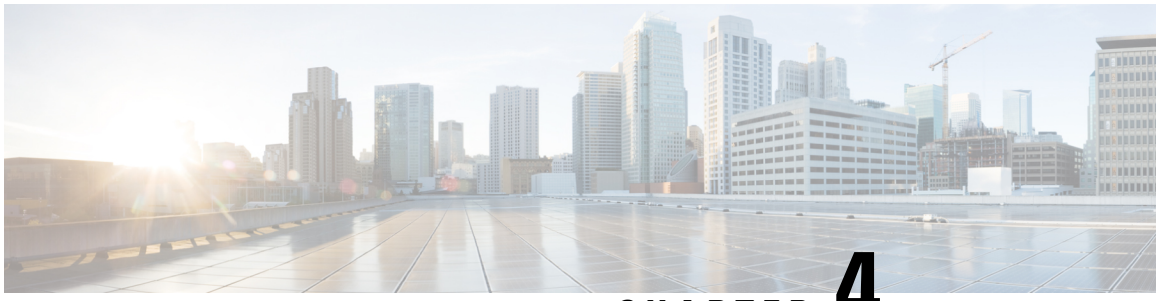
```
telnet 10.106.19.12 2052
ucs-qa-10-A login: ucs-ldap23\blradmin
```

From a Putty client:

- Login as: **ucs-***auth-domain**username*

```
Login as: ucs-example\jsmith
```

Note If the default authentication is set to local, and the console authentication is set to LDAP, you can log in to the fabric interconnect from a Putty client using **ucs-local\admin**, where admin is the name of the local account.



CHAPTER 4

License Management for the ASA

Cisco Smart Software Licensing lets you purchase and manage a pool of licenses centrally. You can easily deploy or retire devices without having to manage each unit's license key. Smart Software Licensing also lets you see your license usage and needs at a glance.

Note This section only applies to ASA logical devices on the Firepower 4100/9300 chassis. For more information on licensing for Firepower Threat Defense logical devices, see the Firepower Management Center Configuration Guide.

About Smart Software Licensing

This section describes how Smart Software Licensing works.

Note This section only applies to ASA logical devices on the Firepower 4100/9300 chassis. For more information on licensing for Firepower Threat Defense logical devices, see the Firepower Management Center Configuration Guide.

Smart Software Licensing for the ASA

For the ASA application on the Firepower 4100/9300 chassis, Smart Software Licensing configuration is split between the Firepower 4100/9300 chassis supervisor and the application.

- Firepower 4100/9300 chassis—Configure all Smart Software Licensing infrastructure in the supervisor, including parameters for communicating with the License Authority. The Firepower 4100/9300 chassis itself does not require any licenses to operate.

Note Inter-chassis clustering requires that you enable the same Smart Licensing method on each chassis in the cluster.

- ASA Application—Configure all license entitlements in the application.

Note Cisco Transport Gateway is not supported on Firepower 4100/9300 security appliances.

Smart Software Manager and Accounts

When you purchase 1 or more licenses for the device, you manage them in the Cisco Smart Software Manager:

https://software.cisco.com/#module/SmartLicensing

The Smart Software Manager lets you create a master account for your organization.

Note If you do not yet have an account, click the link to set up a new account. The Smart Software Manager lets you create a master account for your organization.

By default, your licenses are assigned to the *Default Virtual Account* under your master account. As the account administrator, you can optionally create additional virtual accounts; for example, you can create accounts for regions, departments, or subsidiaries. Multiple virtual accounts let you more easily manage large numbers of licenses and devices.

Offline Management

If your devices do not have Internet access, and cannot register with the License Authority, you can configure offline licensing.

Satellite Server

If your devices cannot access the internet for security reasons, you can optionally install a local Smart Software Manager satellite server as a virtual machine (VM). The satellite provides a subset of Smart Software Manager functionality, and allows you to provide essential licensing services for all your local devices. Only the satellite needs to connect periodically to the main License Authority to sync your license usage. You can sync on a schedule or you can sync manually.

Once you download and deploy the satellite application, you can perform the following functions without sending data to Cisco SSM using the Internet:

- Activate or register a license
- View your company's licenses

- Transfer licenses between company entities

For more information, see the Smart Software Manager satellite installation and configuration guides on Smart Account Manager satellite.

Licenses and Devices Managed per Virtual Account

Licenses and devices are managed per virtual account: only that virtual account's devices can use the licenses assigned to the account. If you need additional licenses, you can transfer an unused license from another virtual account. You can also transfer devices between virtual accounts.

Only the Firepower 4100/9300 chassis registers as a device, while the ASA applications in the chassis request their own licenses. For example, for a Firepower 9300 chassis with 3 security modules, the chassis counts as one device, but the modules use 3 separate licenses.

Evaluation License

The Firepower 4100/9300 chassis supports two types of evaluation license:

- Chassis-level evaluation mode—Before the Firepower 4100/9300 chassis registers with the Licensing Authority, it operates for 90 days (total usage) in evaluation mode. The ASA cannot request specific entitlements in this mode; only default entitlements are enabled. When this period ends, the Firepower 4100/9300 chassis becomes out-of-compliance.
- Entitlement-based evaluation mode—After the Firepower 4100/9300 chassis registers with the Licensing Authority, you can obtain time-based evaluation licenses that can be assigned to the ASA. In the ASA, you request entitlements as usual. When the time-based license expires, you need to either renew the time-based license or obtain a permanent license.

Note You cannot receive an evaluation license for Strong Encryption (3DES/AES); only permanent licenses support this entitlement.

Smart Software Manager Communication

This section describes how your device communicates with the Smart Software Manager.

Device Registration and Tokens

For each virtual account, you can create a registration token. This token is valid for 30 days by default. Enter this token ID plus entitlement levels when you deploy each chassis, or when you register an existing chassis. You can create a new token if an existing token is expired.

At startup after deployment, or after you manually configure these parameters on an existing chassis, the chassis registers with the Cisco License Authority. When the chassis registers with the token, the License Authority issues an ID certificate for communication between the chassis and the License Authority. This certificate is valid for 1 year, although it will be renewed every 6 months.

Periodic Communication with the License Authority

The device communicates with the License Authority every 30 days. If you make changes in the Smart Software Manager, you can refresh the authorization on the device so the change takes place immediately. Or you can wait for the device to communicate as scheduled.

You can optionally configure an HTTP proxy.

The Firepower 4100/9300 chassis must have internet access either directly or through an HTTP proxy at least every 90 days. Normal license communication occurs every 30 days, but with the grace period, your device will operate for up to 90 days without calling home. After the grace period, you must contact the Licensing Authority, or you will not be able to make configuration changes to features requiring special licenses; operation is otherwise unaffected.

Note If your device is unable to communicate with the license authority for one year, the device will enter an unregistered state without strong encryption licenses.

Out-of-Compliance State

The device can become out of compliance in the following situations:

- Over-utilization—When the device uses unavailable licenses.
- License expiration—When a time-based license expires.
- Lack of communication—When the device cannot reach the Licensing Authority for re-authorization.

To verify whether your account is in, or approaching, an Out-of-Compliance state, you must compare the entitlements currently in use by your Firepower 4100/9300 chassis against those in your Smart Account.

In an out-of-compliance state, you will not be able to make configuration changes to features requiring special licenses, but operation is otherwise unaffected. For example, existing contexts over the Standard license limit can continue to run, and you can modify their configuration, but you will not be able to add a *new* context.

Smart Call Home Infrastructure

By default, a Smart Call Home profile exists in the FXOS configuration that specifies the URL for the Licensing Authority. You cannot remove this profile. Note that the only configurable option for the License profile is the destination address URL for the License Authority. Unless directed by Cisco TAC, you should not change the License Authority URL.

Note Cisco Transport Gateway is not supported on Firepower 4100/9300 security appliances.

Prerequisites for Smart Software Licensing

- Note that this chapter only applies to ASA logical devices on the Firepower 4100/9300 chassis. For more information on licensing for Firepower Threat Defense logical devices, see the Firepower Management Center Configuration Guide.

- Create a master account on the Cisco Smart Software Manager:

  https://software.cisco.com/#module/SmartLicensing

  If you do not yet have an account, click the link to set up a new account. The Smart Software Manager lets you create a master account for your organization.
- Purchase 1 or more licenses from the Cisco Commerce Workspace. On the home page, search for your platform in the **Find Products and Solutions** search field. Some licenses are free, but you still need to add them to your Smart Software Licensing account.
- Ensure internet access or HTTP proxy access from the chassis, so the chassis can contact the Licensing Authority.
- Configure a DNS server so the chassis can resolve the name of the Licensing Authority.
- Set the time for the chassis.
- Configure the Smart Software Licensing infrastructure on the Firepower 4100/9300 chassis before you configure the ASA licensing entitlements.

Guidelines for Smart Software Licensing

ASA Guidelines for Failover and Clustering

Each Firepower 4100/9300 chassis must be registered with the License Authority or satellite server. There is no extra cost for secondary units.

Defaults for Smart Software Licensing

The Firepower 4100/9300 chassis default configuration includes a Smart Call Home profile called “SLProfile” that specifies the URL for the Licensing Authority.

```
scope monitoring
  scope callhome
    scope profile SLProfile
      scope destination SLDest
        set address https://tools.cisco.com/its/service/oddce/services/DDCEService
```

Configure Regular Smart Software Licensing

To communicate with the Cisco License Authority, you can optionally configure an HTTP proxy. To register with the License Authority, you must enter the registration token ID on the Firepower 4100/9300 chassis that you obtained from your Smart Software License account.

Procedure

Step 1 (Optional) Configure the HTTP Proxy, on page 20.

Step 2 Register the Firepower Security Appliance with the License Authority, on page 21.

(Optional) Configure the HTTP Proxy

If your network uses an HTTP proxy for Internet access, you must configure the proxy address for Smart Software Licensing. This proxy is also used for Smart Call Home in general.

Note HTTP proxy with authentication is not supported.

Procedure

Step 1 Enable the HTTP proxy:

scope monitoring
scope callhome
set http-proxy-server-enable on

Example:

```
scope monitoring
  scope callhome
    set http-proxy-server-enable on
```

Step 2 Set the proxy URL:

set http-proxy-server-url *url*

where *url* is the http or https address of the proxy server.

Example:

```
set http-proxy-server-url https://10.1.1.1
```

Step 3 Set the port:

set http-proxy-server-port *port*

Example:

```
set http-proxy-server-port 443
```

Step 4 Commit the buffer:

commit-buffer

Register the Firepower Security Appliance with the License Authority

When you register the Firepower 4100/9300 chassis, the License Authority issues an ID certificate for communication between the Firepower 4100/9300 chassis and the License Authority. It also assigns the Firepower 4100/9300 chassis to the appropriate virtual account. Normally, this procedure is a one-time instance. However, you might need to later re-register the Firepower 4100/9300 chassis if the ID certificate expires because of a communication problem, for example.

Procedure

Step 1 In the Smart Software Manager or the Smart Software Manager Satellite, request and copy a registration token for the virtual account to which you want to add this Firepower 4100/9300 chassis.

For more information on how to request a registration token using the Smart Software Manager Satellite, see the Cisco Smart Software Manager Satellite User Guide (https://www.cisco.com/c/en/us/buy/smart-accounts/software-manager-satellite.html).

Step 2 Enter the registration token on the Firepower 4100/9300 chassis:

scope license

register idtoken *id-token*

Example:

```
scope license
  register idtoken ZGFmNWM5NjgtYmNjYS00ZWI3L
WE3NGItMWJkOGExZjIxNGQ0LTE0NjI2NDYx%0AMDIzNT
V8N3R0dXM1Z0NjWkdpR214eFZhMldBOS9CVnNEYnVKM1
g3R3dvemRD%0AY29NQT0%3D%0A
```

Step 3 To later unregister the device, enter:

scope license

deregister

Deregistering the Firepower 4100/9300 chassis removes the device from your account. All license entitlements and certificates on the device are removed. You might want to deregister to free up a license for a new Firepower 4100/9300 chassis. Alternatively, you can remove the device from the Smart Software Manager.

Step 4 To renew the ID certificate and update the entitlements on all security modules, enter:

scope license

scope licdebug

renew

By default, the ID certificate is automatically renewed every 6 months, and the license entitlement is renewed every 30 days. You might want to manually renew the registration for either of these items if you have a limited window for Internet access, or if you make any licensing changes in the Smart Software Manager, for example.

Change Cisco Success Network Enrollment

You enable Cisco Success Network when you register the Firepower 4100/9300 with the Cisco Smart Software Manager. After that, use the following procedure to view or change enrollment status.

Note Cisco Success Network does not work in evaluation mode.

Procedure

Step 1 Enter the system scope.

scope system

Example:

```
Firepower# scope system
Firepower /system #
```

Step 2 Enter the services scope.

scope services

Example:

```
Firepower /system # scope services
Firepower /system/services #
```

Step 3 Enter the telemetry scope.

scope telemetry

Example:

```
Firepower /system/services # scope telemetry
Firepower /system/services/telemetry #
```

Step 4 Enable or disable the Cisco Success Network feature.

{enable | disable}

Example:

```
Firepower /system/services/telemetry # enable
```

Step 5 Verify the Cisco Success Network status in the Firepower 4100/9300 Chassis.

show detail

Example:

Verify that the **Admin State** shows the correct status of Cisco Success Network.

```
Telemetry:
    Admin State: Enabled
    Oper State: Registering
    Error Message:
    Period: 86400
    Current Task: Registering the device for Telemetry
    (FSM-STAGE:sam:dme:CommTelemetryDataExchSeq:RegisterforTelemetry)
```

Example:

Verify that the **Oper State** shows **OK**, which indicates that telemetry data is sent.

```
Telemetry:
    Admin State: Enabled
    Oper State: Ok
    Error Message:
    Period: 86400
    Current Task:
```

Configure a Smart License Satellite Server for the Firepower 4100/9300 chassis

The following procedure shows how to configure the Firepower 4100/9300 chassis to use a Smart License satellite server.

Before you begin

- Complete all prerequisites listed in the Prerequisites for Smart Software Licensing, on page 18.
- Deploy and set up a Smart Software Satellite Server:

  Download the Smart License Satellite OVA file from Cisco.com and install and configure it on a VMwareESXi server. For more information, see the Smart Software Manager satellite Install Guide.
- Verify that the FQDN of the Smart Software Satellite Server can be resolved by your internal DNSserver.
- Verify whether the satellite trustpoint is already present:

  scope security

  show trustpoint

  Note that the trustpoint is added by default in FXOS version 2.4(1) and later. If the trustpoint is not present, you must add one manually using the following steps:

  1. Go to http://www.cisco.com/security/pki/certs/clrca.cer and copy the entire body of the SSL certificate (from "-----BEGIN CERTIFICATE-----" to "-----END CERTIFICATE-----") into a place you can access during configuration.
  2. Enter security mode:

     scope security
  3. Create and name a trusted point:

     create trustpoint *trustpoint_name*
  4. Specify certificate information for the trust point. Note: the certificate must be in Base64 encoded X.509 (CER) format.

     set certchain *certchain*

     For the *certchain* variable, paste the certificate text that you copied in step 1.

If you do not specify certificate information in the command, you are prompted to enter a certificate or a list of trust points defining a certification path to the root certificate authority (CA). On the next line following your input, type **ENDOFBUF** to finish.

5. Commit the configuration:

   commit-buffer

Procedure

Step 1 Set up the satellite server as the callhome destination:

scope monitoring

scope callhome

scope profile SLProfile

scope destination SLDest

set address https://*[FQDN of Satellite server]***/Transportgateway/services/DeviceRequestHandler**

Step 2 Register the Firepower 4100/9300 chassis with the License Authority (see Register the Firepower Security Appliance with the License Authority, on page 21). Note that you must request and copy the registration token from the Smart License Manager satellite.

Monitoring Smart Software Licensing

See the following commands for viewing license status:

- **show license all**

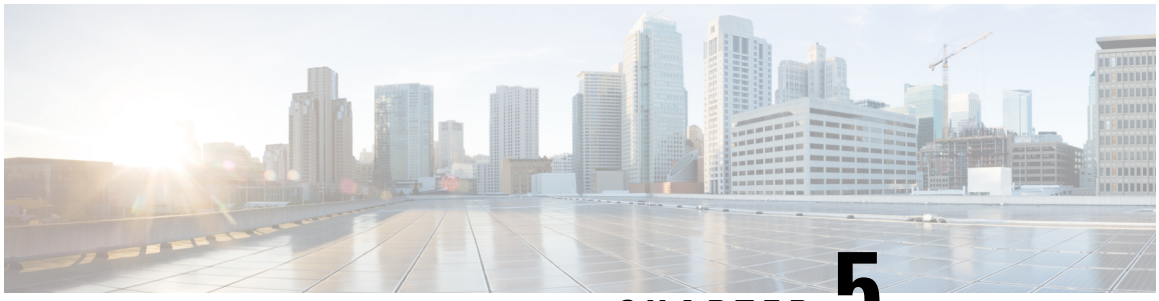
  Displays the state of Smart Software Licensing, Smart Agent version, UDI information, Smart Agent state, global compliance status, the entitlements status, licensing certificate information and schedule Smart Agent tasks.

- **show license status**
- **show license techsupport**

History for Smart Software Licensing

Feature Name	Platform Releases	Description
Cisco Success Network	2.7.1	Cisco Success Network is a user-enabled cloud service. When you enable Cisco Success Network, a secure connection is established between the Firepower 4100/9300 chassis and the Cisco cloud to stream usage information and statistics. Streaming telemetry provides a mechanism that selects data of interest from the ASA and transmits it in a structured format to remote management stations to do the following: • Inform you of available unused features that can improve the effectiveness of the product in your network • Inform you of additional technical support services and monitoring that might be available for your product • Help Cisco improve our products Once you enroll in the Cisco Success Network, the chassis establishes and maintains the secure connection at all times. You can turn off this connection at any time by disabling Cisco Success Network, which disconnects the device from the Cisco Success Network cloud. We introduced the following commands: **scope telemetry {enable | disable}** We introduced the following screens: **System** > **Licensing** > **Cisco Success Network**

Feature Name	Platform Releases	Description
Cisco Smart Software Licensing for the Firepower 4100/9300 chassis	1.1(1)	Smart Software Licensing lets you purchase and manage a pool of licenses. Smart licenses are not tied to a specific serial number. You can easily deploy or retire devices without having to manage each unit's license key. Smart Software Licensing also lets you see your license usage and needs at a glance. Smart Software Licensing configuration is split between the Firepower 4100/9300 chassis supervisor and the security module. We introduced the following commands: **deregister, register idtoken, renew, scope callhome, scope destination, scope licdebug, scope license, scope monitoring, scope profile, set address, set http-proxy-server-enable on, set http-proxy-server-url, set http-proxy-server-port, show license all, show license status, show license techsupport**



CHAPTER 5

User Management

User Accounts

User accounts are used to access the system. You can configure up to 48 local user accounts. Each user account must have a unique username and password.

Admin Account

The admin account is a default user account and cannot be modified or deleted. This account is the system administrator or superuser account and has full privileges. There is no default password assigned to the admin account; you must choose the password during the initial system setup.

The admin account is always active and does not expire. You cannot configure the admin account as inactive.

Locally Authenticated User Accounts

A locally authenticated user account is authenticated directly through the chassis and can be enabled or disabled by anyone with admin or AAA privileges. Once a local user account is disabled, the user cannot log in. Configuration details for disabled local user accounts are not deleted by the database. If you reenable a disabled local user account, the account becomes active again with the existing configuration.

Remotely Authenticated User Accounts

A remotely authenticated user account is any user account that is authenticated through LDAP, RADIUS, or TACACS+.

If a user maintains a local user account and a remote user account simultaneously, the roles defined in the local user account override those maintained in the remote user account.

See the following topics for more information on guidelines for remote authentication, and how to configure and delete remote authentication providers:

- Guidelines for Remote Authentication, on page 29
- Configuring LDAP Providers, on page 88
- Configuring RADIUS Providers, on page 92
- Configuring TACACS+ Providers, on page 95

Expiration of User Accounts

You can configure user accounts to expire at a predefined time. When the expiration time is reached, the user account is disabled.

By default, user accounts do not expire.

After you configure a user account with an expiration date, you cannot reconfigure the account to not expire. You can, however, configure the account with the latest expiration date available.

Guidelines for Usernames

The username is also used as the login ID for Firepower Chassis Manager and the FXOS CLI. When you assign login IDs to user accounts, consider the following guidelines and restrictions:

- The login ID can contain between 1 and 32 characters, including the following:
  - Any alphabetic character
  - Any digit
  - _ (underscore)
  - - (dash)
  - . (dot)
- The login ID must be unique.
- The login ID must start with an alphabetic character. It cannot start with a number or a special character, such as an underscore.

- The login ID is case-sensitive.
- You cannot create an all-numeric login ID.
- After you create a user account, you cannot change the login ID. You must delete the user account and create a new one.

Guidelines for Passwords

A password is required for each locally authenticated user account. A user with admin or AAA privileges can configure the system to perform a password strength check on user passwords. If the password strength check is enabled, each user must have a strong password.

We recommend that each user have a strong password. If you enable the password strength check for locally authenticated users, the Firepower eXtensible Operating System rejects any password that does not meet the following requirements:

- Must contain a minimum of 8 characters and a maximum of 80 characters.
- Must include at least one uppercase alphabetic character.
- Must include at least one lowercase alphabetic character.
- Must include at least one non-alphanumeric (special) character.
- Must not contain a space.
- Must not contain a character that is repeated more than 3 times consecutively, such as aaabbb.
- Must not contain three consecutive numbers or letters in any order, such as passwordABC or password321.
- Must not be identical to the username or the reverse of the username.
- Must pass a password dictionary check. For example, the password must not be based on a standard dictionary word.
- Must not contain the following symbols: $ (dollar sign), ? (question mark), and = (equals sign).

Note This restriction applies whether the password strength check is enabled or not.

- Must not be blank for local user and admin accounts.

Guidelines for Remote Authentication

If a system is configured for one of the supported remote authentication services, you must create a provider for that service to ensure that the Firepower 4100/9300 chassis can communicate with the system. The following guidelines impact user authorization:

User Accounts in Remote Authentication Services

User accounts can exist locally in the Firepower 4100/9300 chassis or in the remote authentication server.

You can view the temporary sessions for users who log in through remote authentication services from the Firepower Chassis Manager or the FXOS CLI.

User Roles in Remote Authentication Services

If you create user accounts in the remote authentication server, you must ensure that the accounts include the roles those users require for working in the Firepower 4100/9300 chassis and that the names of those roles match the names used in FXOS. Based on the role policy, a user might not be allowed to log in, or is granted only read-only privileges.

User Attributes in Remote Authentication Providers

For RADIUS and TACACS+ configurations, you must configure a user attribute for the Firepower 4100/9300 chassis in each remote authentication provider through which users log in to Firepower Chassis Manager or the FXOS CLI. This user attribute holds the roles and locales assigned to each user.

When a user logs in, FXOS does the following:

1. Queries the remote authentication service.
2. Validates the user.
3. If the user is validated, checks the roles and locales assigned to that user.

The following table contains a comparison of the user attribute requirements for the remote authentication providers supported by FXOS:

Authenication Provider	Custom Attribute	Schema Extension	Attribute ID Requirements
LDAP	Optional	You can choose to do one of the following: • Do not extend the LDAP schema and configure an existing, unused attribute that meets the requirements. • Extend the LDAP schema and create a custom attribute with a unique name, such as CiscoAVPair.	The Cisco LDAP implementation requires a unicode type attribute. If you choose to create the CiscoAVPair custom attribute, use the following attribute ID: 1.3.6.1.4.1.9.287247.1 A sample OID is provided in the following section.
RADIUS	Optional	You can choose to do one of the following: • Do not extend the RADIUS schema and use an existing, unused attribute that meets the requirements. • Extend the RADIUS schema and create a custom attribute with a unique name, such as cisco-avpair.	The vendor ID for the Cisco RADIUS implementation is 009 and the vendor ID for the attribute is 001. The following syntax example shows how to specify multiples user roles and locales if you choose to create the cisco-avpair attribute: `shell:roles="admin,aaa" shell:locales="L1,abc"`. Use a comma "," as the delimiter to separate multiple values.

Authenication Provider	Custom Attribute	Schema Extension	Attribute ID Requirements
TACACS+	Required	You must extend the schema and create a custom attribute with the name cisco-av-pair.	The cisco-av-pair name is the string that provides the attribute ID for the TACACS+ provider. The following syntax example shows how to specify multiples user roles and locales when you create the cisco-av-pair attribute: `cisco-av-pair=shell:roles="admin aaa" shell:locales*"L1 abc"`. Using an asterisk (*) in the cisco-av-pair attribute syntax flags the locale as optional, preventing authentication failures for other Cisco devices that use the same authorization profile. Use a space as the delimiter to separate multiple values.

Sample OID for LDAP User Attribute

The following is a sample OID for a custom CiscoAVPair attribute:

```
CN=CiscoAVPair,CN=Schema,
CN=Configuration,CN=X
objectClass: top
objectClass: attributeSchema
cn: CiscoAVPair
distinguishedName: CN=CiscoAVPair,CN=Schema,CN=Configuration,CN=X
instanceType: 0x4
uSNCreated: 26318654
attributeID: 1.3.6.1.4.1.9.287247.1
attributeSyntax: 2.5.5.12
isSingleValued: TRUE
showInAdvancedViewOnly: TRUE
adminDisplayName: CiscoAVPair
adminDescription: UCS User Authorization Field
oMSyntax: 64
lDAPDisplayName: CiscoAVPair
name: CiscoAVPair
objectCategory: CN=Attribute-Schema,CN=Schema,CN=Configuration,CN=X
```

User Roles

The system contains the following user roles:

Administrator

Complete read-and-write access to the entire system. The default admin account is assigned this role by default and it cannot be changed.

Read-Only

Read-only access to system configuration with no privileges to modify the system state.

Password Profile for Locally Authenticated Users

The password profile contains the password history and password change interval properties for all locally authenticated users. You cannot specify a different password profile for each locally authenticated user.

Password History Count

The password history count allows you to prevent locally authenticated users from reusing the same password over and over again. When this property is configured, the Firepower chassis stores passwords that were previously used by locally authenticated users up to a maximum of 15 passwords. The passwords are stored in reverse chronological order with the most recent password first to ensure that the only the oldest password can be reused when the history count threshold is reached.

A user must create and use the number of passwords configured in the password history count before being able to reuse one. For example, if you set the password history count to 8, a locally authenticated user cannot reuse the first password until after the ninth password has expired.

By default, the password history is set to 0. This value disables the history count and allows users to reuse previously passwords at any time.

If necessary, you can clear the password history count for a locally authenticated user and enable reuse of previous passwords.

Password Change Interval

The password change interval enables you to restrict the number of password changes a locally authenticated user can make within a given number of hours. The following table describes the two configuration options for the password change interval.

Interval Configuration	Description	Example
No password change allowed	This option does not allow passwords for locally authenticated users to be changed within a specified number of hours after a password change. You can specify a no change interval between 1 and 745 hours. By default, the no change interval is 24 hours.	For example, to prevent passwords from being changed within 48 hours after a locally authenticated user changes his or her password, set the following: • Change during interval to disable • No change interval to 48

Interval Configuration	Description	Example
Password changes allowed within change interval	This option specifies the maximum number of times that passwords for locally authenticated users can be changed within a pre-defined interval. You can specify a change interval between 1 and 745 hours and a maximum number of password changes between 0 and 10. By default, a locally authenticated user is permitted a maximum of 2 password changes within a 48 hour interval.	For example, to allow a password to be changed a maximum of once within 24 hours after a locally authenticated user changes his or her password, set the following: • Change during interval to enable • Change count to 1 • Change interval to 24

Select the Default Authentication Service

Procedure

Step 1 Enter security mode:

Firepower-chassis # **scope security**

Step 2 Enter default authorization security mode:

Firepower-chassis /security # **scope default-auth**

Step 3 Specify the default authentication:

Firepower-chassis /security/default-auth # **set realm** *auth-type*

where *auth-type* is one of the following keywords:

- **ldap**—Specifies LDAP authentication
- **local**—Specifies local authentication
- **none**—Allows local users to log on without specifying a password
- **radius**—Specifies RADIUS authentication
- **tacacs**—Specifies TACACS+ authentication

Note If Default Authentication and Console Authentication are both set to use the same remote authentication protocol (RADIUS, TACACS+, or LDAP), you cannot change certain aspects of that server's configuration (for example, deleting that server, or changing its order of assignment) without updating these user settings.

Step 4 (Optional) Specify the associated provider group, if any:

Firepower-chassis /security/default-auth # **set auth-server-group** *auth-serv-group-name*

Step 5 (Optional) Specify the maximum amount of time allowed between refresh requests for a user in this domain:

Firepower-chassis /security/default-auth # **set refresh-period** *seconds*

Specify an integer between 60 and 172800. The default is 600 seconds.

If this time limit is exceeded, FXOS considers the web session to be inactive, but it does not terminate the session.

Step 6 (Optional) Specify the maximum amount of time that can elapse after the last refresh request before FXOS considers a web session to have ended:

Firepower-chassis /security/default-auth # **set session-timeout** *seconds*

Specify an integer between 60 and 172800. The default is 7200 seconds.

Note If you set two-factor authentication for a RADIUS or TACACS+ realm, consider increasing the **session-refresh** and **session-timeout** periods so that remote users do not have to reauthenticate too frequently.

Step 7 (Optional) Set the authentication method to two-factor authentication for the realm:

Firepower-chassis /security/default-auth # **set use-2-factor yes**

Note Two-factor authentication applies only to the RADIUS and TACACS+ realms.

Step 8 Commit the transaction to the system configuration:

commit-buffer

Example

The following example sets the default authentication to RADIUS, the default authentication provider group to provider1, enables two-factor authentications, sets the refresh period to 300 seconds (5 minutes), the session timeout period to 540 seconds (9 minutes), and enables two-factor authentication. It then commits the transaction.

```
Firepower-chassis# scope security
Firepower-chassis /security # scope default-auth
Firepower-chassis /security/default-auth # set realm radius
Firepower-chassis /security/default-auth* # set auth-server-group provider1
Firepower-chassis /security/default-auth* # set use-2-factor yes
Firepower-chassis /security/default-auth* # set refresh-period 300
Firepower-chassis /security/default-auth* # set session-timeout 540
Firepower-chassis /security/default-auth* # commit-buffer
Firepower-chassis /security/default-auth #
```

Configuring the Session Timeout

You can use the FXOS CLI to specify the amount of time that can pass without user activity before the Firepower 4100/9300 chassis closes user sessions. You can configure different settings for console sessions and for HTTPS, SSH, and Telnet sessions.

You can set a timeout value up to 3600 seconds (60 minutes). The default value is 600 seconds. To disable this setting, set the session timeout value to 0.

Procedure

Step 1 Enter security mode:

Firepower-chassis # **scope security**

Step 2 Enter default authorization security mode:

Firepower-chassis /security # **scope default-auth**

Step 3 Set the idle timeout for HTTPS, SSH, and Telnet sessions:

Firepower-chassis /security/default-auth # **set session-timeout** *seconds*

Step 4 (Optional) Set the idle timeout for console sessions:

Firepower-chassis /security/default-auth # **set con-session-timeout** *seconds*

Step 5 Commit the transaction to the system configuration:

Firepower-chassis /security/default-auth # **commit-buffer**

Step 6 (Optional) View the session and absolute session timeout settings:

Firepower-chassis /security/default-auth # **show detail**

Example:

```
Default authentication:
 Admin Realm: Local
 Operational Realm: Local
 Web session refresh period(in secs): 600
 Idle Session timeout (in secs) for web, ssh, telnet sessions: 600
 Absolute Session timeout (in secs) for web, ssh, telnet sessions: 3600
 Serial Console Session timeout(in secs): 600
 Serial Console Absolute Session timeout(in secs): 3600
 Admin Authentication server group:
 Operational Authentication server group:
 Use of 2nd factor: No
```

Configuring the Role Policy for Remote Users

By default, read-only access is granted to all users logging in to Firepower Chassis Manager or the FXOS CLI from a remote server using the LDAP, RADIUS, or TACACS+ protocols. For security reasons, it might be desirable to restrict access to those users matching an established user role.

You can configure the role policy for remote users in the following ways:

assign-default-role

When a user attempts to log in and the remote authentication provider does not supply a user role with the authentication information, the user is allowed to log in with a read-only user role.

This is the default behavior.

no-login

When a user attempts to log in and the remote authentication provider does not supply a user role with the authentication information, access is denied.

Procedure

Step 1 Enter security mode:

Firepower-chassis # **scope security**

Step 2 Specify whether user access to Firepower Chassis Manager and the FXOS CLI should be restricted based on user roles:

Firepower-chassis /security # **set remote-user default-role** {**assign-default-role** | **no-login**}

Step 3 Commit the transaction to the system configuration:

Firepower-chassis /security # **commit-buffer**

Example

The following example sets the role policy for remote users and commits the transaction:

```
Firepower-chassis# scope security
Firepower-chassis /security # set remote-user default-role no-login
Firepower-chassis /security* # commit-buffer
Firepower-chassis /security #
```

Enabling Password Strength Check for Locally Authenticated Users

If the password strength check is enabled, the Firepower eXtensible Operating System does not permit a user to choose a password that does not meet the guidelines for a strong password (see Guidelines for Passwords, on page 29).

Procedure

Step 1 Enter security mode:

Firepower-chassis # **scope security**

Step 2 Specify whether the password strength check is enabled or disabled:

Firepower-chassis /security # **set enforce-strong-password** {**yes** | **no**}

Example

The following example enables the password strength check:

```
Firepower-chassis# scope security
Firepower-chassis /security # set enforce-strong-password yes
Firepower-chassis /security* # commit-buffer
Firepower-chassis /security #
```

Set the Maximum Number of Login Attempts

You can configure the maximum number of failed login attempts allowed before a user is locked out of the Firepower 4100/9300 chassis for a specified amount of time. If a user exceeds the set maximum number of login attempts, the user is locked out of the system. No notification appears indicating that the user is locked out. In this event, the user must wait the specified amount of time before attempting to log in.

Perform these steps to configure the maximum number of login attempts.

Note

-
- The default maximum number of unsuccessful login attempts is 3. The default amount of time the user is locked out of the system after exceeding the maximum number of login attemps is 60 minutes (3600 seconds).

Procedure

Step 1 From the FXOS CLI, enter security mode:

scope security

Step 2 Set the maximum number of unsuccessful login attempts.

set max-login-attempts *num_attempts*

Step 3 Specify the amount of time (in seconds) the user should remain locked out of the system after reaching the maximum number of login attempts:

set user-account-unlock-time

unlock_time

Step 4 Commit the configuration:

commit-buffer

Configuring the Maximum Number of Password Changes for a Change Interval

Procedure

Step 1 Enter security mode:

Firepower-chassis # **scope security**

Step 2 Enter password profile security mode:

Firepower-chassis /security # **scope password-profile**

Step 3 Restrict the number of password changes a locally authenticated user can make within a given number of hours:

Firepower-chassis /security/password-profile # **set change-during-interval enable**

Step 4 Specify the maximum number of times a locally authenticated user can change his or her password during the Change Interval:

Firepower-chassis /security/password-profile # **set change-count** *pass-change-num*

This value can be anywhere from 0 to 10.

Step 5 Specify the maximum number of hours over which the number of password changes specified in the **Change Count** field are enforced:

Firepower-chassis /security/password-profile # **set change-interval** *num-of-hours*

This value can be anywhere from 1 to 745 hours.

For example, if this field is set to 48 and the **Change Count** field is set to 2, a locally authenticated user can make no more than 2 password changes within a 48 hour period.

Step 6 Commit the transaction to the system configuration:

Firepower-chassis /security/password-profile # **commit-buffer**

Example

The following example enables the change during interval option, sets the change count to 5, sets the change interval to 72 hours, and commits the transaction:

```
Firepower-chassis # scope security
Firepower-chassis /security # scope password-profile
Firepower-chassis /security/password-profile # set change-during-interval enable
Firepower-chassis /security/password-profile* # set change-count 5
Firepower-chassis /security/password-profile* # set change-interval 72
Firepower-chassis /security/password-profile* # commit-buffer
Firepower-chassis /security/password-profile #
```

Configuring a No Change Interval for Passwords

Procedure

Step 1 Enter security mode:

Firepower-chassis # **scope security**

Step 2 Enter password profile security mode:

Firepower-chassis /security # **scope password-profile**

Step 3 Disable the change during interval feature:

Firepower-chassis /security/password-profile # **set change-during-interval disable**

Step 4 Specify the minimum number of hours that a locally authenticated user must wait before changing a newly created password:

Firepower-chassis /security/password-profile # **set no-change-interval** *min-num-hours*

This value can be anywhere from 1 to 745 hours.

This interval is ignored if the **Change During Interval** property is not set to **Disable**.

Step 5 Commit the transaction to the system configuration:

Firepower-chassis /security/password-profile # **commit-buffer**

Example

The following example disables the change during interval option, sets the no change interval to 72 hours, and commits the transaction:

```
Firepower-chassis # scope security
Firepower-chassis /security # scope password-profile
Firepower-chassis /security/password-profile # set change-during-interval disable
Firepower-chassis /security/password-profile* # set no-change-interval 72
Firepower-chassis /security/password-profile* # commit-buffer
Firepower-chassis /security/password-profile #
```

Configuring the Password History Count

Procedure

Step 1 Enter security mode:

Firepower-chassis # **scope security**

Step 2 Enter password profile security mode:

Firepower-chassis /security # **scope password-profile**

Step 3 Specify the number of unique passwords that a locally authenticated user must create before that user can reuse a previously used password:

Firepower-chassis /security/password-profile # **set history-count** *num-of-passwords*

This value can be anywhere from 0 to 15.

By default, the **History Count** field is set to 0, which disables the history count and allows users to reuse previously used passwords at any time.

Step 4 Commit the transaction to the system configuration:

Firepower-chassis /security/password-profile # **commit-buffer**

Example

The following example configures the password history count and commits the transaction:

```
Firepower-chassis # scope security
Firepower-chassis /security # scope password-profile
Firepower-chassis /security/password-profile # set history-count 5
Firepower-chassis /security/password-profile* # commit-buffer
Firepower-chassis /security/password-profile #
```

Creating a Local User Account

Procedure

Step 1 Enter security mode:

Firepower-chassis# **scope security**

Step 2 Create the user account:

Firepower-chassis /security # **create local-user** *local-user-name*

where *local-user-name* is the account name to be used when logging into this account. This name must be unique and meet the guidelines and restrictions for user account names (see Guidelines for Usernames, on page 28).

After you create the user, the login ID cannot be changed. You must delete the user account and create a new one.

Step 3 Specify whether the local user account is enabled or disabled:

Firepower-chassis /security/local-user # **set account-status** {**active**| **inactive**}

Step 4 Set the password for the user account:

Firepower-chassis /security/local-user # **set password**

Enter a password: *password*

Confirm the password: *password*

If password strength check is enabled, a user's password must be strong and the Firepower eXtensible Operating System rejects any password that does not meet the strength check requirements (see Guidelines for Passwords, on page 29).

Note Passwords must not contain the following symbols: $ (dollar sign), ? (question mark), and = (equals sign). This restriction applies whether the password strength check is enabled or not.

Step 5 (Optional) Specify the first name of the user:

Firepower-chassis /security/local-user # **set firstname** *first-name*

Step 6 (Optional) Specify the last name of the user:

Firepower-chassis /security/local-user # **set lastname** *last-name*

Step 7 (Optional) Specify the date that the user account expires. The *month* argument is the first three letters of the month name.

Firepower-chassis /security/local-user # **set expiration** *month day-of-month year*

Note After you configure a user account with an expiration date, you cannot reconfigure the account to not expire. You can, however, configure the account with the latest expiration date available.

Step 8 (Optional) Specify the user e-mail address.

Firepower-chassis /security/local-user # **set email** *email-addr*

Step 9 (Optional) Specify the user phone number.

Firepower-chassis /security/local-user # **set phone** *phone-num*

Step 10 (Optional) Specify the SSH key used for passwordless access.

Firepower-chassis /security/local-user # **set sshkey** *ssh-key*

Step 11 All users are assigned the *read-only* role by default and this role cannot be removed. For each additional role that you want to assign to the user:

Firepower-chassis /security/local-user # **create role** *role-name*

where *role-name* is the role that represents the privileges you want to assign to the user account (see User Roles, on page 31).

Note Changes in user roles and privileges do not take effect until the next time the user logs in. If a user is logged in when you assign a new role to or remove an existing role from a user account, the active session continues with the previous roles and privileges.

Step 12 To remove an assigned role from the user:

Firepower-chassis /security/local-user # **delete role** *role-name*

All users are assigned the *read-only* role by default and this role cannot be removed.

Step 13 Commit the transaction.

Firepower-chassis security/local-user # **commit-buffer**

Example

The following example creates the user account named kikipopo, enables the user account, sets the password to foo12345, assigns the admin user role, and commits the transaction:

```
Firepower-chassis# scope security
Firepower-chassis /security # create local-user kikipopo
Firepower-chassis /security/local-user* # set account-status active
Firepower-chassis /security/local-user* # set password
Enter a password:
Confirm the password:
Firepower-chassis /security/local-user* # create role admin
Firepower-chassis /security/local-user* # commit-buffer
Firepower-chassis /security/local-user #
```

The following example creates the user account named lincey, enables the user account, sets an OpenSSH key for passwordless access, assigns the aaa and operations user roles, and commits the transaction.

```
Firepower-chassis# scope security
Firepower-chassis /security # create local-user lincey
Firepower-chassis /security/local-user* # set account-status active
Firepower-chassis /security/local-user* #  set sshkey "ssh-rsa
AAAAB3NzaC1yc2EAAAABIwAAAIEAuo9VQ2CmWBI9/S1f30klCWjnV3lgdXMzO0WUl5iPw85lkdQqap+NFuNmHcb4K
iaQB8X/PDdmtlxQQcawclj+k8f4VcOelBxlsGk5luq5ls1ob1VOIEwcKEL/h5lrdbNlI8y3SS9I/gGiBZ9ARlop9LDpD
m8HPh2LOgyH7Ei1MI8="
Firepower-chassis /security/local-user* # create role aaa
Firepower-chassis /security/local-user* # create role operations
Firepower-chassis /security/local-user* # commit-buffer
Firepower-chassis /security/local-user #
```

The following example creates the user account named jforlenz, enables the user account, sets a Secure SSH key for passwordless access, and commits the transaction.

```
Firepower-chassis# scope security
Firepower-chassis /security # create local-user jforlenz
Firepower-chassis /security/local-user* # set account-status active
Firepower-chassis /security/local-user* #  set sshkey
Enter lines one at a time. Enter ENDOFBUF to finish. Press ^C to abort.
User's SSH key:
> ---- BEGIN SSH2 PUBLIC KEY ----
>AAAAB3NzaC1yc2EAAAABIwAAAIEAuo9VQ2CmWBI9/S1f30klCWjnV3lgdXMzO0WUl5iPw8
>5lkdQqap+NFuNmHcb4KiaQB8X/PDdmtlxQQcawclj+k8f4VcOelBxlsGk5luq5ls1ob1VO
>IEwcKEL/h5lrdbNlI8y3SS9I/gGiBZ9ARlop9LDpDm8HPh2LOgyH7Ei1MI8=
> ---- END SSH2 PUBLIC KEY ----
> ENDOFBUF
Firepower-chassis /security/local-user* # commit-buffer
Firepower-chassis /security/local-user #
```

Deleting a Local User Account

Procedure

Step 1 Enter security mode:

Firepower-chassis# **scope security**

Step 2 Delete the local-user account:

Firepower-chassis /security # **delete local-user** *local-user-name*

Step 3 Commit the transaction to the system configuration:

Firepower-chassis /security # **commit-buffer**

Example

The following example deletes the foo user account and commits the transaction:

```
Firepower-chassis# scope security
Firepower-chassis /security # delete local-user foo
Firepower-chassis /security* # commit-buffer
Firepower-chassis /security #
```

Activating or Deactivating a Local User Account

You must be a user with admin or AAA privileges to activate or deactivate a local user account.

Procedure

Step 1 Enter security mode:

Firepower-chassis# **scope security**

Step 2 Enter local-user security mode for the user you want to activate or deactivate:

Firepower-chassis /security # **scope local-user** *local-user-name*

Step 3 Specify whether the local user account is active or inactive:

Firepower-chassis /security/local-user # **set account-status** {**active** | **inactive**}

Note The admin user account is always set to active. It cannot be modified.

Step 4 Commit the transaction to the system configuration:

Firepower-chassis /security/local-user # **commit-buffer**

Example

The following example enables a local user account called accounting:

```
Firepower-chassis# scope security
Firepower-chassis /security # scope local-user accounting
Firepower-chassis /security* # commit-buffer
Firepower-chassis /security #
```

Clearing the Password History for a Locally Authenticated User

Procedure

Step 1 Enter security mode:

Firepower-chassis # **scope security**

Step 2 Enter local user security mode for the specified user account:

Firepower-chassis /security # **scope local-user** *user-name*

Step 3 Clear the password history for the specified user account:

Firepower-chassis /security/local-user # **clear password-history**

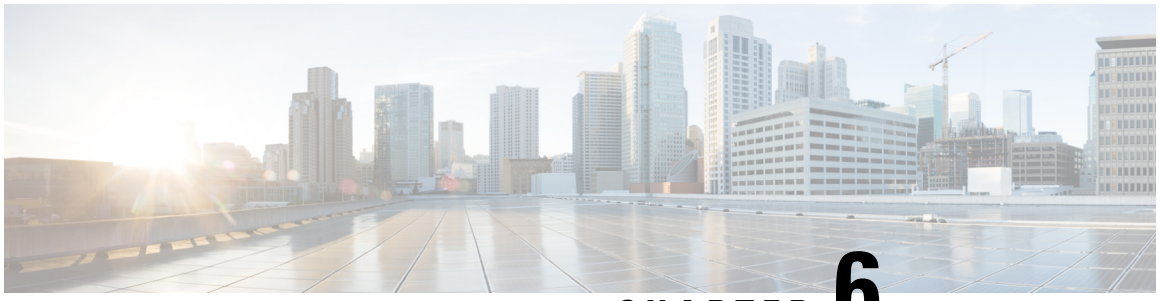
Step 4 Commit the transaction to the system configuration:

Firepower-chassis /security/local-user # **commit-buffer**

Example

The following example clears the password history and commits the transaction:

```
Firepower-chassis # scope security
Firepower-chassis /security # scope local-user admin
Firepower-chassis /security/local-user # clear password-history
Firepower-chassis /security/local-user* # commit-buffer
Firepower-chassis /security/local-user #
```



CHAPTER 6

Image Management

About Image Management

The Firepower 4100/9300 chassis uses two basic types of images:

Note All images are digitally signed and validated through Secure Boot. Do not modify the image in any way or you will receive a validation error.

- Platform Bundle—The Firepower platform bundle is a collection of multiple independent images that operate on the Firepower Supervisor and Firepower security module/engine. The platform bundle is a Firepower eXtensible Operating System software package.
- Application—Application images are the software images you want to deploy on the security module/engine of the Firepower 4100/9300 chassis. Application images are delivered as Cisco Secure Package files (CSP) and are stored on the supervisor until deployed to a security module/engine as part of logical device creation or in preparation for later logical device creation. You can have multiple different versions of the same application image type stored on the Firepower Supervisor.

Note If you are upgrading both the Platform Bundle image and one or more Application images, you must upgrade the Platform Bundle first.

Downloading Images from Cisco.com

Download FXOS and application images from Cisco.com so you can upload them to the Firepower chassis.

Before you begin

You must have a Cisco.com account.

Procedure

Step 1 Using a web browser, navigate to http://www.cisco.com/go/firepower9300-software or http://www.cisco.com/go/firepower4100-software.
The software download page for the Firepower 4100/9300 chassis is opened in the browser.

Step 2 Find and then download the appropriate software image to your local computer.

Downloading a Firepower eXtensible Operating System Software Image to the Firepower 4100/9300 chassis

You can use FTP, SCP, SFTP, or TFTP to copy the FXOS software image to the Firepower 4100/9300 chassis.

Before you begin

Collect the following information that you will need to import a configuration file:

- IP address and authentication credentials for the server from which you are copying the image
- Fully qualified name of the FXOS image file

Procedure

Step 1 Enter firmware mode:

Firepower-chassis # **scope firmware**

Step 2 Download the FXOS software image:

Firepower-chassis /firmware # **download image** *URL*

Specify the URL for the file being imported using one of the following syntax:

- **ftp**://*username@hostname*/*path*/*image_name*
- **scp**://*username@hostname*/*path*/*image_name*
- **sftp**://*username@hostname*/*path*/*image_name*
- **tftp**://*hostname***:***port-num*/*path*/*image_name*

Step 3 To monitor the download process:

Firepower-chassis /firmware # **show package** *image_name* **detail**

Example

The following example copies an image using the SCP protocol:

```
Firepower-chassis # scope firmware
Firepower-chassis /firmware # download image
scp://user@192.168.1.1/images/fxos-k9.1.1.1.119.SPA
Firepower-chassis /firmware # show package fxos-k9.1.1.1.119.SPA detail
Download task:
    File Name: fxos-k9.1.1.1.119.SPA
    Protocol: scp
    Server: 192.168.1.1
    Userid:
    Path:
    Downloaded Image Size (KB): 5120
    State: Downloading
    Current Task: downloading image fxos-k9.1.1.1.119.SPA from
192.168.1.1(FSM-STAGE:sam:dme:FirmwareDownloaderDownload:Local)
```

Verifying the Integrity of an Image

The integrity of the image is automatically verified when a new image is added to the Firepower 4100/9300 chassis. If needed, you can use the following procedure to manually verify the integrity of an image.

Procedure

Step 1 Connect to the FXOS CLI (see Accessing the FXOS CLI, on page 12).

Step 2 Enter firmware mode:

Firepower-chassis# **scope firmware**

Step 3 List images:

Firepower-chassis /firmware # **show package**

Step 4 Verify the image:

Firepower-chassis /firmware # **verify platform-pack version** *version_number*

version_number is the version number of the FXOS platform bundle you are verifying--for example, 1.1(2.51).

Step 5 The system will warn you that verification could take several minutes.

Enter **yes** to confirm that you want to proceed with verification.

Step 6 To check the status of the image verification:

Firepower-chassis /firmware # **show validate-task**

Upgrading the Firepower eXtensible Operating System Platform Bundle

Before you begin

Download the platform bundle software image from Cisco.com (see Downloading Images from Cisco.com, on page 46) and then download that image to the Firepower 4100/9300 chassis (see Downloading a Logical Device Software Image to the Firepower 4100/9300 chassis, on page 49).

Note The upgrade process typically takes between 20 and 30 minutes.

If you are upgrading a Firepower 9300 or Firepower 4100 Series security appliance that is running a standalone logical device or if you are upgrading a Firepower 9300 security appliance that is running an intra-chassis cluster, traffic will not traverse through the device while it is upgrading.

If you are upgrading Firepower 9300 or a Firepower 4100 Series security appliance that is part of an inter-chassis cluster, traffic will not traverse through the device being upgraded while it is upgrading. However, the other devices in the cluster will continue to pass traffic.

Procedure

Step 1 Connect to the FXOS CLI (see Accessing the FXOS CLI, on page 12).

Step 2 Enter firmware mode:

Firepower-chassis# **scope firmware**

Step 3 Enter auto-install mode:

Firepower-chassis /firmware # **scope auto-install**

Step 4 Install the FXOS platform bundle:

Firepower-chassis /firmware/auto-install # **install platform platform-vers** *version_number*

version_number is the version number of the FXOS platform bundle you are installing--for example, 1.1(2.51).

Step 5 The system will first verify the software package that you want to install. It will inform you of any incompatibility between currently installed applications and the specified FXOS platform software package. It will also warn you that any existing sessions will be terminated and that the system will need to be rebooted as part of the upgrade.

Enter **yes** to confirm that you want to proceed with verification.

Step 6 Enter **yes** to confirm that you want to proceed with installation, or enter **no** to cancel the installation.

The Firepower eXtensible Operating System unpacks the bundle and upgrades/reloads the components.

Step 7 To monitor the upgrade process:

a) Enter **scope firmware**.
b) Enter **scope auto-install**.
c) Enter **show fsm status expand**.

Downloading a Logical Device Software Image to the Firepower 4100/9300 chassis

You can use FTP, SCP, SFTP, or TFTP to copy the logical device software image to the Firepower 4100/9300 chassis.

Before you begin

Collect the following information that you will need to import a configuration file:

- IP address and authentication credentials for the server from which you are copying the image
- Fully qualified name of the software image file

Procedure

Step 1 Enter Security Services mode:

Firepower-chassis # **scope ssa**

Step 2 Enter Application Software mode:

Firepower-chassis /ssa # **scope app-software**

Step 3 Download the logical device software image:

Firepower-chassis /ssa/app-software # **download image** *URL*

Specify the URL for the file being imported using one of the following syntax:

- **ftp**://*username@hostname*/*path*
- **scp**://*username@hostname*/*path*
- **sftp**://*username@hostname*/*path*
- **tftp**://*hostname*:*port-num*/*path*

Step 4 To monitor the download process:

Firepower-chassis /ssa/app-software # **show download-task**

Step 5 To view the downloaded applications:

Firepower-chassis /ssa/app-software # **up**

Firepower-chassis /ssa # **show app**

Step 6 To view details for a specific application:

Firepower-chassis /ssa # **scope app** *application_type image_version*

Firepower-chassis /ssa/app # **show expand**

Example

The following example copies an image using the SCP protocol:

```
Firepower-chassis # scope ssa
Firepower-chassis /ssa # scope app-software
Firepower-chassis /ssa/app-software # download image
scp://user@192.168.1.1/images/cisco-asa.9.4.1.65.csp
Firepower-chassis /ssa/app-software # show download-task

Downloads for Application Software:
    File Name                     Protocol   Server               Userid          State
    ----------------------------- ---------- -------------------- --------------- -----
    cisco-asa.9.4.1.65.csp        Scp        192.168.1.1          user            Downloaded

Firepower-chassis /ssa/app-software # up

Firepower-chassis /ssa # show app

Application:
    Name       Version    Description Author     Deploy Type CSP Type    Is Default App
    ---------- ---------- ----------- ---------- ----------- ----------- --------------
    asa        9.4.1.41   N/A                    Native      Application No
    asa        9.4.1.65   N/A                    Native      Application Yes

Firepower-chassis /ssa # scope app asa 9.4.1.65
Firepower-chassis /ssa/app # show expand

Application:
    Name: asa
    Version: 9.4.1.65
    Description: N/A
    Author:
    Deploy Type: Native
    CSP Type: Application
    Is Default App: Yes

    App Attribute Key for the Application:
        App Attribute Key Description
        ----------------- -----------
        cluster-role      This is the role of the blade in the cluster
        mgmt-ip           This is the IP for the management interface
        mgmt-url          This is the management URL for this application

    Net Mgmt Bootstrap Key for the Application:
        Bootstrap Key Key Data Type Is the Key Secret Description
        ------------- ------------- ----------------- -----------
        PASSWORD      String        Yes               The admin user password.

    Port Requirement for the Application:
        Port Type: Data
        Max Ports: 120
```

```
        Min Ports: 1

        Port Type: Mgmt
        Max Ports: 1
        Min Ports: 1

        Mgmt Port Sub Type for the Application:
            Management Sub Type
            -------------------
            Default

        Port Type: Cluster
        Max Ports: 1
        Min Ports: 0
Firepower-chassis /ssa/app #
```

Updating the Image Version for a Logical Device

Use this procedure to upgrade the ASA application image to a new version, or set the Firepower Threat Defense application image to a new startup version that will be used in a disaster recovery scenario.

After initial creation of a FTD logical device, you do not upgrade the FTD logical device using Firepower Chassis Manager or the FXOS CLI. To upgrade a FTD logical device, you must use Firepower Management Center. See the Firepower System Release Notes for more information: http://www.cisco.com/c/en/us/support/security/defense-center/products-release-notes-list.html.

Also, note that any updates to the FTD logical device will not be reflected on the **Logical Devices** > **Edit** and **System** > **Updates** pages in Firepower Chassis Manager. On these pages, the version shown indicates the software version (CSP image) that was used to create the FTD logical device.

When you change the startup version on an ASA logical device, the ASA upgrades to that version and all configuration is restored. Use the following workflows to change the ASA startup version, depending on your configuration:

ASA High Availability -

1. Change the logical device image version(s) on the standby unit.
2. Make the standby unit active.
3. Change the application version(s) on the other unit.

ASA Inter-Chassis Cluster -

1. Change the startup version on the data unit.
2. Make the data unit the control unit.
3. Change the startup version on the original control unit (now data).

Before you begin

Download the application image you want to use for the logical device from Cisco.com (see Downloading Images from Cisco.com, on page 46) and then download that image to the Firepower 4100/9300 chassis (see Downloading a Logical Device Software Image to the Firepower 4100/9300 chassis, on page 49).

If you are upgrading both the Platform Bundle image and one or more Application images, you must upgrade the Platform Bundle first.

Procedure

Step 1 Enter Security Services mode:

Firepower-chassis # **scope ssa**

Step 2 Set the scope to the security module you are updating:

Firepower-chassis /ssa # **scope slot** *slot_number*

Step 3 Set the scope to the application you are updating:

Firepower-chassis /ssa/slot # **scope app-instance** *app_template*

Step 4 Set the Startup version:

Firepower-chassis /ssa/slot/app-instance # **set startup-version** *version_number*

Step 5 Commit the configuration:

commit-buffer

Commits the transaction to the system configuration. The application image is updated and the application restarts.

Example

The following example updates the software image for an ASA running on security module 1. Notice that you can use the **show** command to view the update status.

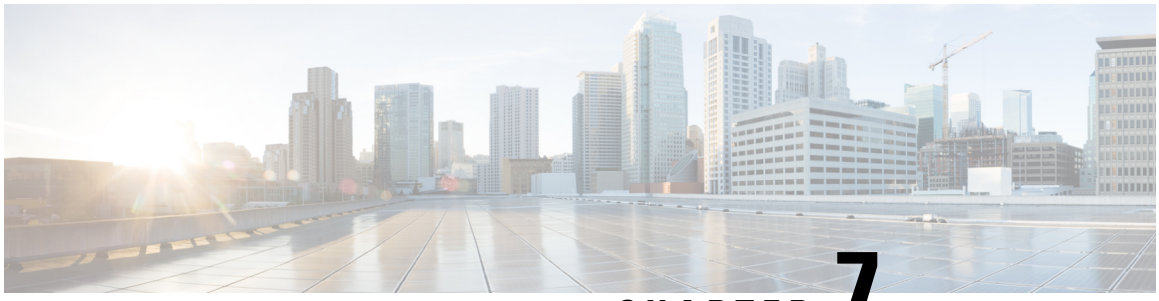
```
Firepower-chassis# scope ssa
Firepower-chassis /ssa # scope slot 1
Firepower-chassis /ssa/slot # scope app-instance asa
Firepower-chassis /ssa/slot/app-instance # set startup-version 9.4.1.65
Firepower-chassis /ssa/slot/app-instance* # show configuration pending
 enter app-instance asa
+    set startup-version 9.4.1.65
 exit
Firepower-chassis /ssa/slot/app-instance* # commit-buffer
Firepower-chassis /ssa/slot/app-instance # show

Application Instance:
    Application Name Admin State Operational State Running Version Startup Version
    ---------------- ----------- ----------------- --------------- ---------------
    asa              Enabled     Updating          9.4.1.41        9.4.1.65
Firepower-chassis /ssa/slot/app-instance #
Firepower-chassis /ssa/slot/app-instance # show

Application Instance:
    Application Name Admin State Operational State Running Version Startup Version
    ---------------- ----------- ----------------- --------------- ---------------
    asa              Enabled     Online            9.4.1.65        9.4.1.65
Firepower-chassis /ssa/slot/app-instance #
```

Firmware Upgrade

For information about upgrading the firmware on your Firepower 4100/9300 chassis, see the *Cisco Firepower 4100/9300 FXOS Firmware Upgrade Guide*.



CHAPTER 7

Platform Settings

Changing the Management IP Address

Before you begin

You can change the management IP address on the Firepower 4100/9300 chassis from the FXOS CLI.

Note After changing the management IP address, you will need to reestablish any connections to Firepower Chassis Manager or the FXOS CLI using the new address.

Procedure

Step 1 Connect to the FXOS CLI (see Accessing the FXOS CLI, on page 12).

Step 2 To configure an IPv4 management IP address:

a) Set the scope for fabric-interconnect a:

Firepower-chassis# **scope fabric-interconnect a**

b) To view the current management IP address, enter the following command:

Firepower-chassis /fabric-interconnect # **show**

c) Enter the following command to configure a new management IP address and gateway:

Firepower-chassis /fabric-interconnect # **set out-of-band ip** *ip_address* **netmask** *network_mask* **gw** *gateway_ip_address*

d) Commit the transaction to the system configuration:

Firepower-chassis /fabric-interconnect* # **commit-buffer**

Step 3 To configure an IPv6 management IP address:

a) Set the scope for fabric-interconnect a:

Firepower-chassis# **scope fabric-interconnect a**

b) Set the scope for management IPv6 configuration:

Firepower-chassis /fabric-interconnect # **scope ipv6-config**

c) To view the current management IPv6 address, enter the following command:

Firepower-chassis /fabric-interconnect/ipv6-config # **show ipv6-if**

d) Enter the following command to configure a new management IP address and gateway:

Firepower-chassis /fabric-interconnect/ipv6-config # **set out-of-band ipv6** *ipv6_address* **ipv6-prefix** *prefix_length* **ipv6-gw** *gateway_address*

Note Only IPv6 Global Unicast addresses are supported as the chassis's IPv6 management address.

e) Commit the transaction to the system configuration:

Firepower-chassis /fabric-interconnect/ipv6-config* # **commit-buffer**

Example

The following example configures an IPv4 management interface and gateway:

```
Firepower-chassis# scope fabric-interconnect a
Firepower-chassis /fabric-interconnect # show

Fabric Interconnect:
    ID   OOB IP Addr     OOB Gateway     OOB Netmask     OOB IPv6 Address OOB IPv6 Gateway
 Prefix Operability
    ---- --------------- --------------- --------------- ---------------- ----------------
 ------ -----------
    A    192.0.2.112     192.0.2.1       255.255.255.0   ::               ::
 64     Operable
Firepower-chassis /fabric-interconnect # set out-of-band ip 192.0.2.111 netmask 255.255.255.0
 gw 192.0.2.1
Warning: When committed, this change may disconnect the current CLI session
Firepower-chassis /fabric-interconnect* #commit-buffer
Firepower-chassis /fabric-interconnect #
```

The following example configures an IPv6 management interface and gateway:

```
Firepower-chassis# scope fabric-interconnect a
Firepower-chassis /fabric-interconnect # scope ipv6-config
Firepower-chassis /fabric-interconnect/ipv6-config # show ipv6-if

Management IPv6 Interface:
    IPv6 Address                      Prefix     IPv6 Gateway
    --------------------------------- ---------- ------------
    2001::8998                        64         2001::1
Firepower-chassis /fabric-interconnect/ipv6-config # set out-of-band ipv6 2001::8999
ipv6-prefix 64 ipv6-gw 2001::1
Firepower-chassis /fabric-interconnect/ipv6-config* # commit-buffer
Firepower-chassis /fabric-interconnect/ipv6-config #
```

Setting the Date and Time

Use the CLI commands described below to configure the network time protocol (NTP) on the system, to set the date and time manually, or to view the current system time.

NTP settings are automatically synced between the Firepower 4100/9300 chassis and any logical devices installed on the chassis.

Note If you are deploying Firepower Threat Defense on the Firepower 4100/9300 chassis, you must configure NTP on the Firepower 4100/9300 chassis so that Smart Licensing will work properly and to ensure proper timestamps on device registrations. You should use the same NTP server for both the Firepower 4100/9300 chassis and the Firepower Management Center, but note that you cannot use Firepower Management Center as the NTP server for the Firepower 4100/9300 chassis.

Setting the Time Zone

Procedure

Step 1 Enter system mode:

Firepower-chassis# **scope system**

Step 2 Enter system services mode:

Firepower-chassis /system # **scope services**

Step 3 Set the time zone:

Firepower-chassis /system/services # **set timezone**

At this point, you are prompted to enter a number corresponding to your continent, country, and time zone region. Enter the appropriate information at each prompt.

When you have finished specifying the location information, you are prompted to confirm that the correct time zone information is being set. Enter **1** (yes) to confirm, or **2** (no) to cancel the operation.

Step 4 To view the configured time zone:

Firepower-chassis /system/services # **top**

Firepower-chassis# **show timezone**

Example

The following example configures the time zone to the Pacific time zone region, commits the transaction, and displays the configured time zone:

```
Firepower-chassis# scope system
Firepower-chassis /system # scope services
Firepower-chassis /system/services # set timezone
Please identify a location so that time zone rules can be set correctly.
Please select a continent or ocean.
1) Africa            4) Arctic Ocean     7) Australia       10) Pacific Ocean
2) Americas          5) Asia             8) Europe
3) Antarctica        6) Atlantic Ocean   9) Indian Ocean
#? 2
Please select a country.
 1) Anguilla                28) Haiti
 2) Antigua & Barbuda       29) Honduras
 3) Argentina               30) Jamaica
 4) Aruba                   31) Martinique
 5) Bahamas                 32) Mexico
 6) Barbados                33) Montserrat
 7) Belize                  34) Nicaragua
 8) Bolivia                 35) Panama
 9) Brazil                  36) Paraguay
10) Canada                  37) Peru
11) Caribbean Netherlands   38) Puerto Rico
12) Cayman Islands          39) St Barthelemy
13) Chile                   40) St Kitts & Nevis
14) Colombia                41) St Lucia
15) Costa Rica              42) St Maarten (Dutch part)
16) Cuba                    43) St Martin (French part)
17) Curacao                 44) St Pierre & Miquelon
18) Dominica                45) St Vincent
19) Dominican Republic      46) Suriname
20) Ecuador                 47) Trinidad & Tobago
21) El Salvador             48) Turks & Caicos Is
22) French Guiana           49) United States
23) Greenland               50) Uruguay
24) Grenada                 51) Venezuela
25) Guadeloupe              52) Virgin Islands (UK)
26) Guatemala               53) Virgin Islands (US)
27) Guyana
#? 49
Please select one of the following time zone regions.
 1) Eastern Time
 2) Eastern Time - Michigan - most locations
 3) Eastern Time - Kentucky - Louisville area
 4) Eastern Time - Kentucky - Wayne County
 5) Eastern Time - Indiana - most locations
 6) Eastern Time - Indiana - Daviess, Dubois, Knox & Martin Counties
 7) Eastern Time - Indiana - Pulaski County
 8) Eastern Time - Indiana - Crawford County
 9) Eastern Time - Indiana - Pike County
10) Eastern Time - Indiana - Switzerland County
11) Central Time
12) Central Time - Indiana - Perry County
```

```
13) Central Time - Indiana - Starke County
14) Central Time - Michigan - Dickinson, Gogebic, Iron & Menominee Counties
15) Central Time - North Dakota - Oliver County
16) Central Time - North Dakota - Morton County (except Mandan area)
17) Central Time - North Dakota - Mercer County
18) Mountain Time
19) Mountain Time - south Idaho & east Oregon
20) Mountain Standard Time - Arizona (except Navajo)
21) Pacific Time
22) Pacific Standard Time - Annette Island, Alaska
23) Alaska Time
24) Alaska Time - Alaska panhandle
25) Alaska Time - southeast Alaska panhandle
26) Alaska Time - Alaska panhandle neck
27) Alaska Time - west Alaska
28) Aleutian Islands
29) Hawaii
#? 21

The following information has been given:

        United States
        Pacific Time

Therefore timezone 'America/Los_Angeles' will be set.
Local time is now:      Wed Jun 24 07:39:25 PDT 2015.
Universal Time is now:  Wed Jun 24 14:39:25 UTC 2015.
Is the above information OK?
1) Yes
2) No
#? 1
Firepower-chassis /system/services* # commit-buffer
Firepower-chassis /system/services # top
Firepower-chassis# show timezone
Timezone: America/Los_Angeles (Pacific Time)
Firepower-chassis#
```

Setting the Date and Time Using NTP

NTP is used to implement a hierarchical system of servers that provide a precisely synchronized time among network systems. This kind of accuracy is required for time-sensitive operations, such as validating CRLs, which include a precise time stamp.

Before you begin

If you use a hostname for the NTP server, you must configure a DNS server. See Configuring DNS Servers, on page 102.

Procedure

Step 1 Enter system mode:

Firepower-chassis# **scope system**

Step 2 Enter system services mode:

Firepower-chassis /system # **scope services**

Step 3 Configure the system to use the NTP server with the specified hostname, IPv4, or IPv6 address:

Firepower-chassis /system/services # **create ntp-server** *{hostname | ip-addr | ip6-addr}*

Step 4 Commit the transaction to the system configuration:

Firepower-chassis /system/services # **commit-buffer**

Step 5 To view the synchronization status for all configured NTP servers:

Firepower-chassis /system/services # **show ntp-server**

Step 6 To view the synchronization status for a specific NTP server:

Firepower-chassis /system/services # **scope ntp-server** *{hostname | ip-addr | ip6-addr}*

Firepower-chassis /system/services/ntp-server # **show detail**

Example

The following example configures an NTP server with the IP address 192.168.200.101 and commits the transaction:

```
Firepower-chassis# scope system
Firepower-chassis /system # scope services
Firepower-chassis /system/services # create ntp-server 192.168.200.101
Firepower-chassis /system/services* # commit-buffer
Firepower-chassis /system/services #
```

The following example configures an NTP server with the IPv6 address 4001::6 and commits the transaction:

```
Firepower-chassis# scope system
Firepower-chassis /system # scope services
Firepower-chassis /system/services # create ntp-server 4001::6
Firepower-chassis /system/services* # commit-buffer
Firepower-chassis /system/services #
```

Deleting an NTP Server

Procedure

Step 1 Enter system mode:

Firepower-chassis# **scope system**

Step 2 Enter system services mode:

Firepower-chassis /system # **scope services**

Step 3 Delete the NTP server with the specified hostname, IPv4, or IPv6 address:

Firepower-chassis /system/services # **delete ntp-server** *{hostname | ip-addr | ip6-addr}*

Step 4 Commit the transaction to the system configuration:

Firepower-chassis /system/services # **commit-buffer**

Example

The following example deletes the NTP server with the IP address 192.168.200.101 and commits the transaction:

```
Firepower-chassis# scope system
Firepower-chassis /system # scope services
Firepower-chassis /system/services # delete ntp-server 192.168.200.101
Firepower-chassis /system/services* # commit-buffer
Firepower-chassis /system/services #
```

The following example deletes the NTP server with the IPv6 address 4001::6 and commits the transaction:

```
Firepower-chassis# scope system
Firepower-chassis /system # scope services
Firepower-chassis /system/services # delete ntp-server 4001::6
Firepower-chassis /system/services* # commit-buffer
Firepower-chassis /system/services #
```

Setting the Date and Time Manually

This section describes how to set the date and time manually on the Firepower chassis. System clock modifications take effect immediately.

Note If the system clock is currently being synchronized with an NTP server, you will not be able to set the date and time manually.

Procedure

Step 1 Enter system mode:

Firepower-chassis# **scope system**

Step 2 Enter system services mode:

Firepower-chassis /system # **scope services**

Step 3 Configure the system clock:

Firepower-chassis /system/services # **set clock** *month day year hour min sec*

For month, use the first three digits of the month. Hours must be entered using the 24-hour format, where 7 pm would be entered as 19.

System clock modifications take effect immediately. You do not need to commit the buffer.

Example

The following example configures the system clock:

```
Firepower-chassis# scope system
Firepower-chassis /system # scope services
Firepower-chassis /system/services # set clock jun 24 2015 15 27 00
Firepower-chassis /system/services #
```

Configuring SSH

The following procedure describes how to enable or disable SSH access to the Firepower chassis, how to enable the FXOS chassis as an SSH client, and how to configure the various algorithms used by SSH for encryption, key exchange, and message authentication for both the SSH server and SSH client.

SSH is enabled by default.

Procedure

Step 1 Enter system mode:

Firepower-chassis # **scope system**

Step 2 Enter system services mode:

Firepower-chassis /system # **scope services**

Step 3 To configure SSH access to the Firepower chassis, do one of the following:

- To allow SSH access to the Firepower chassis, enter the following command:

  Firepower-chassis /system/services # **enable ssh-server**

- To disallow SSH access to the Firepower chassis, enter the following command:

  Firepower-chassis /system/services # **disable ssh-server**

Step 4 Commit the transaction to the system configuration:

Firepower /system/services # **commit-buffer**

Example

The following example enables SSH access to the Firepower chassis and commits the transaction:

```
Firepower# scope system
Firepower /system # scope services
Firepower /system/services # enable ssh-server
```

```
Firepower /system/services* # commit-buffer
Firepower /system/services #
```

Configuring Telnet

The following procedure describes how to enable or disable Telnet access to the Firepower chassis. Telnet is disabled by default.

Note Telnet configuration is currently only available using the CLI.

Procedure

Step 1 Enter system mode:

Firepower-chassis # **scope system**

Step 2 Enter system services mode:

Firepower-chassis /system # **scope services**

Step 3 To configure Telnet access to the Firepower chassis, do one of the following:

- To allow Telnet access to the Firepower chassis, enter the following command:

  Firepower-chassis /system/services # **enable telnet-server**

- To disallow Telnet access to the Firepower chassis, enter the following command:

  Firepower-chassis /system/services # **disable telnet-server**

Step 4 Commit the transaction to the system configuration:

Firepower /system/services # **commit-buffer**

Example

The following example enables Telnet and commits the transaction:

```
Firepower-chassis# scope system
Firepower-chassis /system # scope services
Firepower-chassis /services # enable telnet-server
Firepower-chassis /services* # commit-buffer
Firepower-chassis /services #
```

Configuring SNMP

This section describes how to configure the Simple Network Management Protocol (SNMP) on the Firepower chassis. See the following topics for more information:

About SNMP

The Simple Network Management Protocol (SNMP) is an application-layer protocol that provides a message format for communication between SNMP managers and agents. SNMP provides a standardized framework and a common language used for the monitoring and management of devices in a network.

The SNMP framework consists of three parts:

- An SNMP manager—The system used to control and monitor the activities of network devices using SNMP.
- An SNMP agent—The software component within the Firepower chassis that maintains the data for the Firepower chassis and reports the data, as needed, to the SNMP manager. The Firepower chassis includes the agent and a collection of MIBs. To enable the SNMP agent and create the relationship between the manager and agent, enable and configure SNMP in the Firepower Chassis Manager or the FXOS CLI.
- A managed information base (MIB)—The collection of managed objects on the SNMP agent.

The Firepower chassis supports SNMPv1, SNMPv2c and SNMPv3. Both SNMPv1 and SNMPv2c use a community-based form of security. SNMP is defined in the following:

- RFC 3410 (http://tools.ietf.org/html/rfc3410)
- RFC 3411 (http://tools.ietf.org/html/rfc3411)
- RFC 3412 (http://tools.ietf.org/html/rfc3412)
- RFC 3413 (http://tools.ietf.org/html/rfc3413)
- RFC 3414 (http://tools.ietf.org/html/rfc3414)
- RFC 3415 (http://tools.ietf.org/html/rfc3415)
- RFC 3416 (http://tools.ietf.org/html/rfc3416)
- RFC 3417 (http://tools.ietf.org/html/rfc3417)
- RFC 3418 (http://tools.ietf.org/html/rfc3418)
- RFC 3584 (http://tools.ietf.org/html/rfc3584)

Note Be aware that SNMP versions 1 and 2c have serious known security issues: they transmit all information without encryption, including the community string, which serves as the only form of authentication in these versions.

SNMP Notifications

A key feature of SNMP is the ability to generate notifications from an SNMP agent. These notifications do not require that requests be sent from the SNMP manager. Notifications can indicate improper user authentication, restarts, the closing of a connection, loss of connection to a neighbor router, or other significant events.

The Firepower chassis generates SNMP notifications as either traps or informs. Traps are less reliable than informs because the SNMP manager does not send any acknowledgment when it receives a trap, and the Firepower chassis cannot determine if the trap was received. An SNMP manager that receives an inform request acknowledges the message with an SNMP response protocol data unit (PDU). If the Firepower chassis does not receive the PDU, it can send the inform request again.

However, informs are available only with SNMPv2c, which is considered insecure, and is not recommended.

SNMP Security Levels and Privileges

SNMPv1, SNMPv2c, and SNMPv3 each represent a different security model. The security model combines with the selected security level to determine the security mechanism applied when the SNMP message is processed.

The security level determines the privileges required to view the message associated with an SNMP trap. The privilege level determines whether the message needs to be protected from disclosure or authenticated. The supported security level depends upon which security model is implemented. SNMP security levels support one or more of the following privileges:

- noAuthNoPriv—No authentication or encryption
- authNoPriv—Authentication but no encryption
- authPriv—Authentication and encryption

SNMPv3 provides for both security models and security levels. A security model is an authentication strategy that is set up for a user and the role in which the user resides. A security level is the permitted level of security within a security model. A combination of a security model and a security level determines which security mechanism is employed when handling an SNMP packet.

Supported Combinations of SNMP Security Models and Levels

The following table identifies what the combinations of security models and levels mean.

Table 4: SNMP Security Models and Levels

Model	Level	Authentication	Encryption	What Happens
v1	noAuthNoPriv	Community string	No	Uses a community string match for authentication.
v2c	noAuthNoPriv	Community string	No	Uses a community string match for authentication.

Model	Level	Authentication	Encryption	What Happens
v3	noAuthNoPriv	Username	No	Uses a username match for authentication. **Note** While you can configure it, FXOS does not support use of `noAuthNoPriv` with SNMP version 3.
v3	authNoPriv	HMAC-SHA	No	Provides authentication based on the HMAC Secure Hash Algorithm (SHA).
v3	authPriv	HMAC-SHA	DES	Provides authentication based on the HMAC-SHA algorithm. Provides Data Encryption Standard (DES) 56-bit encryption in addition to authentication based on the Cipher Block Chaining (CBC) DES (DES-56) standard.

SNMPv3 Security Features

SNMPv3 provides secure access to devices by a combination of authenticating and encrypting frames over the network. SNMPv3 authorizes management operations only by configured users and encrypts SNMP messages. The SNMPv3 User-Based Security Model (USM) refers to SNMP message-level security and offers the following services:

- Message integrity—Ensures that messages have not been altered or destroyed in an unauthorized manner and that data sequences have not been altered to an extent greater than can occur non-maliciously.
- Message origin authentication—Ensures that the claimed identity of the user on whose behalf received data was originated is confirmed.
- Message confidentiality and encryption—Ensures that information is not made available or disclosed to unauthorized individuals, entities, or processes.

SNMP Support

The Firepower chassis provides the following support for SNMP:

Support for MIBs

The Firepower chassis supports read-only access to MIBs.

For information about the specific MIBs available and where you can obtain them, see the Cisco FXOS MIB Reference Guide.

Authentication Protocol for SNMPv3 Users

The Firepower chassis supports the HMAC-SHA-96 (SHA) authentication protocol for SNMPv3 users.

AES Privacy Protocol for SNMPv3 Users

The Firepower chassis uses Advanced Encryption Standard (AES) as one of the privacy protocols for SNMPv3 message encryption and conforms with RFC 3826.

The privacy password, or priv option, offers a choice of DES or 128-bit AES encryption for SNMP security encryption. If you enable AES-128 configuration and include a privacy password for an SNMPv3 user, the Firepower chassis uses the privacy password to generate a 128-bit AES key. The AES privacy password can have a minimum of eight characters. If the passphrases are specified in clear text, you can specify a maximum of 64 characters.

Enabling SNMP and Configuring SNMP Properties

Procedure

Step 1 Enter monitoring mode:

Firepower-chassis# **scope monitoring**

Step 2 Enable SNMP:

Firepower-chassis /monitoring # **enable snmp**

Step 3 (Optional) Enter SNMP community mode:

Firepower-chassis /monitoring # **set snmp community**

After you enter the **set snmp community** command, you are prompted to enter the SNMP community name.

When you specify an SNMP community name, you are also automatically enabling SNMP versions 1 and 2c for polling requests from the SNMP remote manager.

Note Be aware that SNMP versions 1 and 2c have serious known security issues: they transmit all information without encryption, including the community string, which serves as the only form of authentication in these versions.

Step 4 Specify the SNMP community name; this community name is used as a SNMP password. The community name can be any alphanumeric string up to 32 characters.

Firepower-chassis /monitoring # **Enter a snmp community:** *community-name*

There can be only one community name; however, you can use **set snmp community** to overwrite the existing name. To delete an existing community name (also disabling SNMP versions 1 and 2c for polling requests from the SNMP remote manager), enter **set snmp community** but do not type a community string; that is, simply press **Enter** again. After you commit the buffer, **show snmp** output will include the line `Is Community Set: No`.

Step 5 Specify the system contact person responsible for SNMP. The system contact name can be any alphanumeric string up to 255 characters, such as an email address or name and telephone number.

Firepower-chassis /monitoring # **set snmp syscontact** *system-contact-name*

Step 6 Specify the location of the host on which the SNMP agent (server) runs. The system location name can be any alphanumeric string up to 512 characters.

Firepower-chassis /monitoring # **set snmp syslocation** *system-location-name*

Step 7 Commit the transaction to the system configuration:

Firepower-chassis /monitoring # **commit-buffer**

Example

The following example enables SNMP, configures an SNMP community named SnmpCommSystem2, configures a system contact named contactperson, configures a contact location named systemlocation, and commits the transaction:

```
Firepower-chassis# scope monitoring
Firepower-chassis /monitoring # enable snmp
Firepower-chassis /monitoring* # set snmp community
Enter a snmp community: SnmpCommSystem2
Firepower-chassis /monitoring* # set snmp syscontact contactperson1
Firepower-chassis /monitoring* # set snmp syslocation systemlocation
Firepower-chassis /monitoring* # commit-buffer
Firepower-chassis /monitoring #
```

What to do next

Create SNMP traps and users.

Creating an SNMP Trap

The following procedure describes how to create SNMP traps.

Note You can define up to eight SNMP traps.

Procedure

Step 1 Enter monitoring mode:

Firepower-chassis# **scope monitoring**

Step 2 Enable SNMP:

Firepower-chassis /monitoring # **enable snmp**

Step 3 Create an SNMP trap with the specified host name, IPv4 address, or IPv6 address.

Firepower-chassis /monitoring # **create snmp-trap** {*hostname* | *ip-addr* | *ip6-addr*}

Step 4 Specify the SNMP community string, or version 3 user name, to be used with the SNMP trap:

Firepower-chassis /monitoring/snmp-trap # **set community** *community-name*

Specifies the SNMPv1/v2c community string, or the SNMPv3 user name, to permit access to the trap destination. You are queried for the community name after you enter this command. The name can be up to 32 characters with no spaces; the name is not displayed as you type.

Step 5 Specify the port to be used for the SNMP trap:

Firepower-chassis /monitoring/snmp-trap # **set port** *port-num*

Step 6 Specify the SNMP version and model used for the trap:

Firepower-chassis /monitoring/snmp-trap # **set version** {**v1** | **v2c** | **v3**}

Note Be aware that SNMP versions 1 and 2c have serious known security issues: they transmit all information without encryption, including the community string, which serves as the only form of authentication in these versions.

Step 7 (Optional) Specify the type of trap to send.

Firepower-chassis /monitoring/snmp-trap # **set notificationtype** {**traps** | **informs**}

This can be:

- **traps** if you select v2c or v3 for the version.
- **informs** if you select v2c for the version.

Note An inform notification can be sent only if you select v2c for the version.

Step 8 (Optional) If you select v3 for the version, specify the privilege associated with the trap:

Firepower-chassis /monitoring/snmp-trap # **set v3privilege** {**auth** | **noauth** | **priv**}

This can be:

- **auth**—Authentication but no encryption.
- **noauth**—No authentication or encryption. Note that while you can specify it, FXOS does not support this security level with SNMPv3.
- **priv**—Authentication and encryption.

Step 9 Commit the transaction to the system configuration:

Firepower-chassis /monitoring/snmp-trap # **commit-buffer**

Example

The following example enables SNMP, creates an SNMP trap using an IPv4 address, specifies that the trap will use the SnmpCommSystem2 community on port 2, sets the version to v3, sets the notification type to traps, sets the v3 privilege to priv, and commits the transaction:

```
Firepower-chassis# scope monitoring
Firepower-chassis /monitoring # enable snmp
Firepower-chassis /monitoring* # create snmp-trap 192.168.100.112
Firepower-chassis /monitoring/snmp-trap* # set community SnmpCommSystem2
Firepower-chassis /monitoring/snmp-trap* # set port 2
Firepower-chassis /monitoring/snmp-trap* # set version v3
Firepower-chassis /monitoring/snmp-trap* # set notificationtype traps
Firepower-chassis /monitoring/snmp-trap* # set v3privilege priv
Firepower-chassis /monitoring/snmp-trap* # commit-buffer
Firepower-chassis /monitoring/snmp-trap #
```

The following example enables SNMP, creates an SNMP trap using an IPv6 address, specifies that the trap will use the SnmpCommSystem3 community on port 2, sets the version to v3, sets the notification type to traps, sets the v3 privilege to priv, and commits the transaction:

```
Firepower-chassis# scope monitoring
Firepower-chassis /monitoring # enable snmp
Firepower-chassis /monitoring* # create snmp-trap 2001::1
Firepower-chassis /monitoring/snmp-trap* # set community SnmpCommSystem3
Firepower-chassis /monitoring/snmp-trap* # set port 2
Firepower-chassis /monitoring/snmp-trap* # set version v3
Firepower-chassis /monitoring/snmp-trap* # set notificationtype traps
Firepower-chassis /monitoring/snmp-trap* # set v3privilege priv
Firepower-chassis /monitoring/snmp-trap* # commit-buffer
Firepower-chassis /monitoring/snmp-trap #
```

Deleting an SNMP Trap

Procedure

Step 1 Enter monitoring mode:

Firepower-chassis# **scope monitoring**

Step 2 Delete the SNMP trap with the specified hostname or IP address:

Firepower-chassis /monitoring # **delete snmp-trap** {*hostname* | *ip-addr*}

Step 3 Commit the transaction to the system configuration:

Firepower-chassis /monitoring # **commit-buffer**

Example

The following example deletes the SNMP trap at IP address 192.168.100.112 and commits the transaction:

```
Firepower-chassis# scope monitoring
Firepower-chassis /monitoring # delete snmp-trap 192.168.100.112
Firepower-chassis /monitoring* # commit-buffer
Firepower-chassis /monitoring #
```

Creating an SNMPv3 User

Procedure

Step 1 Enter monitoring mode:

Firepower-chassis# **scope monitoring**

Step 2 Enable SNMP:

Firepower-chassis /monitoring # **enable snmp**

Step 3 Create an SNMPv3 user:

Firepower-chassis /monitoring # **create snmp-user** *user-name*

After you enter the **create snmp-user** command, you are prompted to enter a password.

The Firepower eXtensible Operating System rejects any password that does not meet the following requirements:

- Must contain a minimum of 8 characters and a maximum of 80 characters.
- Must contain only letters, numbers, and the following characters:

  ~`!@#%^&*()_-+{}[]|\:;"'<,>./
- Must not contain the following symbols: $ (dollar sign), ? (question mark), or = (equals sign).
- Must contain at least five different characters.
- Must not contain too many consecutively incrementing or decrementing numbers or letters. For example, the string "12345" has four such characters, and the string "ZYXW" has three. If the total number of such characters exceeds a certain limit (typically more than around 4-6 such occurrences), the simplicity check will fail.

  Note The consecutively incrementing or decrementing character count is not reset when non-incrementing or decrementing characters are used in between. For example, abcd&!21 will fail the password check, but abcd&!25, will not.

Step 4 Enable or disable the use of AES-128 encryption:

Firepower-chassis /monitoring/snmp-user # **set aes-128** {**no** | **yes**}

By default, AES-128 encryption is disabled.

Step 5 Specify the user privacy password:

Firepower-chassis /monitoring/snmp-user # **set priv-password**

After you enter the **set priv-password** command, you are prompted to enter and confirm the privacy password.

The Firepower eXtensible Operating System rejects any password that does not meet the following requirements:

- Must contain a minimum of 8 characters and a maximum of 80 characters.
- Must contain only letters, numbers, and the following characters:

  ~`!@#%^&*()_-+{}[]|\:;"'<,>./
- Must not contain the following symbols: $ (dollar sign), ? (question mark), or = (equals sign).
- Must contain at least five different characters.
- Must not contain too many consecutively incrementing or decrementing numbers or letters. For example, the string "12345" has four such characters, and the string "ZYXW" has three. If the total number of such characters exceeds a certain limit (typically more than around 4-6 such occurrences), the simplicity check will fail.

Note The consecutively incrementing or decrementing character count is not reset when non-incrementing or decrementing characters are used in between. For example, abcd&!21 will fail the password check, but abcd&!25, will not.

Step 6 Commit the transaction to the system configuration:

Firepower-chassis /monitoring/snmp-user # **commit-buffer**

Example

The following example enables SNMP, creates an SNMPv3 user named snmp-user14, enables AES-128 encryption, sets the password and privacy password, and commits the transaction:

```
Firepower-chassis# scope monitoring
Firepower-chassis /monitoring # enable snmp
Firepower-chassis /monitoring* # create snmp-user snmp-user14
Password:
Firepower-chassis /monitoring/snmp-user* # set aes-128 yes
Firepower-chassis /monitoring/snmp-user* # set priv-password
Enter a password:
Confirm the password:
Firepower-chassis /monitoring/snmp-user* # commit-buffer
Firepower-chassis /monitoring/snmp-user #
```

Deleting an SNMPv3 User

Procedure

Step 1 Enter monitoring mode:

Firepower-chassis# **scope monitoring**

Step 2 Delete the specified SNMPv3 user:

Firepower-chassis /monitoring # **delete snmp-user** *user-name*

Step 3 Commit the transaction to the system configuration:

Firepower-chassis /monitoring # **commit-buffer**

Example

The following example deletes the SNMPv3 user named snmp-user14 and commits the transaction:

```
Firepower-chassis# scope monitoring
Firepower-chassis /monitoring # delete snmp-user snmp-user14
Firepower-chassis /monitoring* # commit-buffer
Firepower-chassis /monitoring #
```

Viewing Current SNMP Settings

Use the following CLI commands to display current SNMP settings, users and traps.

Procedure

Step 1 Enter monitoring mode:

firepower# **scope monitoring**

Step 2 Display the current SNMP settings:

firepower/monitoring # **show snmp**

```
Name: snmp
    Admin State: Enabled
    Port: 161
    Is Community Set: Yes
    Sys Contact: R_Admin
    Sys Location:
```

Step 3 List the currently defined SNMPv3 users:

firepower/monitoring # **show snmp-user**

```
SNMPv3 User:
    Name                      Authentication type
    ------------------------- -------------------
    snmp-user1                Sha
    testuser                  Sha
    snmp-user2                Sha
```

Step 4 List the currently defined SNMP traps:

firepower/monitoring # **show snmp-trap**

```
SNMP Trap:
    SNMP Trap                 Port     Community  Version V3 Privilege Notification Type
    ------------------------- -------- ---------- ------- ------------ -----------------
    trap1_informs             162      ****       V2c     Noauth       Informs
    192.168.10.100            162      ****       V3      Noauth       Traps
```

Example

This example show how to display detailed information about a specific SNMPv3 user:

```
firepower /monitoring # show snmp-user snmp-user1 detail

SNMPv3 User:
    Name: snmp-user1
    Authentication type: Sha
    Password: ****
    Privacy password: ****
    Use AES-128: Yes
```

```
firepower /monitoring #
```

Configuring HTTPS

This section describes how to configure HTTPS on the Firepower 4100/9300 chassis.

Note You can change the HTTPS port using Firepower Chassis Manager or the FXOS CLI. All other HTTPS configuration can only be done using the FXOS CLI.

Certificates, Key Rings, and Trusted Points

HTTPS uses components of the Public Key Infrastructure (PKI) to establish secure communications between two devices, such as a client's browser and the Firepower 4100/9300 chassis.

Encryption Keys and Key Rings

Each PKI device holds a pair of asymmetric Rivest-Shamir-Adleman (RSA) encryption keys, one kept private and one made public, stored in an internal key ring. A message encrypted with either key can be decrypted with the other key. To send an encrypted message, the sender encrypts the message with the receiver's public key, and the receiver decrypts the message using its own private key. A sender can also prove its ownership of a public key by encrypting (also called 'signing') a known message with its own private key. If a receiver can successfully decrypt the message using the public key in question, the sender's possession of the corresponding private key is proven. Encryption keys can vary in length, with typical lengths from 512 bits to 2048 bits. In general, a longer key is more secure than a shorter key. FXOS provides a default key ring with an initial 2048-bit key pair, and allows you to create additional key rings.

The default key ring certificate must be manually regenerated if the cluster name changes or the certificate expires.

Certificates

To prepare for secure communications, two devices first exchange their digital certificates. A certificate is a file containing a device's public key along with signed information about the device's identity. To merely support encrypted communications, a device can generate its own key pair and its own self-signed certificate. When a remote user connects to a device that presents a self-signed certificate, the user has no easy method to verify the identity of the device, and the user's browser will initially display an authentication warning. By default, FXOS contains a built-in self-signed certificate containing the public key from the default key ring.

Trusted Points

To provide stronger authentication for FXOS, you can obtain and install a third-party certificate from a trusted source, or trusted point, that affirms the identity of your device. The third-party certificate is signed by the issuing trusted point, which can be a root certificate authority (CA) or an intermediate CA or trust anchor that is part of a trust chain that leads to a root CA. To obtain a new certificate, you must generate a certificate request through FXOS and submit the request to a trusted point.

Important The certificate must be in Base64 encoded X.509 (CER) format.

Creating a Key Ring

FXOS supports a maximum of 8 key rings, including the default key ring.

Procedure

Step 1 Enter security mode:

Firepower-chassis # **scope security**

Step 2 Create and name the key ring:

Firepower-chassis # **create keyring** *keyring-name*

Step 3 Set the SSL key length in bits:

Firepower-chassis # **set modulus** {**mod1024** | **mod1536** | **mod2048** | **mod512**}

Step 4 Commit the transaction:

Firepower-chassis # **commit-buffer**

Example

The following example creates a keyring with a key size of 1024 bits:

```
Firepower-chassis# scope security
Firepower-chassis /security # create keyring kr220
Firepower-chassis /security/keyring* # set modulus mod1024
Firepower-chassis /security/keyring* # commit-buffer
Firepower-chassis /security/keyring #
```

What to do next

Create a certificate request for this key ring.

Regenerating the Default Key Ring

The default key ring certificate must be manually regenerated if the cluster name changes or the certificate expires.

Procedure

Step 1 Enter security mode:

Firepower-chassis # **scope security**

Step 2 Enter key ring security mode for the default key ring:

Firepower-chassis /security # **scope keyring default**

Step 3 Regenerate the default key ring:

Firepower-chassis /security/keyring # **set regenerate yes**

Step 4 Commit the transaction:

Firepower-chassis # **commit-buffer**

Example

The following example regenerates the default key ring:

```
Firepower-chassis# scope security
Firepower-chassis /security # scope keyring default
Firepower-chassis /security/keyring* # set regenerate yes
Firepower-chassis /security/keyring* # commit-buffer
Firepower-chassis /security/keyring #
```

Creating a Certificate Request for a Key Ring

Creating a Certificate Request for a Key Ring with Basic Options

Procedure

Step 1 Enter security mode:

Firepower-chassis # **scope security**

Step 2 Enter configuration mode for the key ring:

Firepower-chassis /security # **scope keyring** *keyring-name*

Step 3 Create a certificate request using the IPv4 or IPv6 address specified, or the name of the fabric interconnect. You are prompted to enter a password for the certificate request.

Firepower-chassis /security/keyring # **create certreq** {**ip** [*ipv4-addr* | *ipv6-v6*] |**subject-name** *name*}

Step 4 Commit the transaction:

Firepower-chassis /security/keyring/certreq # **commit-buffer**

Step 5 Display the certificate request, which you can copy and send to a trust anchor or certificate authority:

Firepower-chassis /security/keyring # **show certreq**

Example

The following example creates and displays a certificate request with an IPv4 address for a key ring, with basic options:

```
Firepower-chassis# scope security
Firepower-chassis /security # scope keyring kr220
Firepower-chassis /security/keyring # create certreq ip 192.168.200.123 subject-name sjc04
Certificate request password:
Confirm certificate request password:
Firepower-chassis /security/keyring* # commit-buffer
Firepower-chassis /security/keyring # show certreq
Certificate request subject name: sjc04
Certificate request ip address: 192.168.200.123
Certificate request e-mail name:
Certificate request country name:
State, province or county (full name):
Locality (eg, city):
Organization name (eg, company):
Organization Unit name (eg, section):
Request:
-----BEGIN CERTIFICATE REQUEST-----
MIIBfTCB5wIBADARMQ8wDQYDVQQDEwZzYW1jMDQwgZ8wDQYJKoZIhvcNAQEBBQAD
gY0AMIGJAoGBALpKn1t8qMZO4UGqILKFXQQc2c8b/vW2rnRF8OPhKbhghLA1YZ1F
JqcYEG5Yl1+vgohLBTd45s0GC8m4RTLJWHo4SwccAUXQ5Zngf45YtX1WsylwUWV4
0re/zgTk/WCd56RfOBvWR2Dtztu2pGA14sd761zLxt29K7R8mzj6CAUVAgMBAAGg
LTArBgkqhkiG9w0BCQ4xHjAcMBoGA1UdEQEB/wQQMA6CBnNhbWMwNIcECsEiXjAN
BgkqhkiG9w0BAQQFAAOBgQCsxN0qUHYGFoQw56RwQueLTNPnrndqUwuZHUO03Teg
nhsyu4satpyiPqVV9viKZ+spvc6x5PWIcTWgHhH8BimOb/0OKuG8kwfIGGsEDlAv
TTYvUP+BZ9OFiPbRIA718S+V8ndXr1HejiQGxlDNqoN+odCXPc5kjoXD0lZTL09H
BA==
-----END CERTIFICATE REQUEST-----

Firepower-chassis /security/keyring #
```

What to do next

- Copy the text of the certificate request, including the BEGIN and END lines, and save it in a file. Send the file with the certificate request to a trust anchor or certificate authority to obtain a certificate for the key ring.
- Create a trusted point and set the certificate chain for the certificate of trust received from the trust anchor.

Creating a Certificate Request for a Key Ring with Advanced Options

Procedure

Step 1 Enter security mode:

Firepower-chassis # **scope security**

Step 2 Enter configuration mode for the key ring:

Firepower-chassis /security # **scope keyring** *keyring-name*

Step 3 Create a certificate request:

Firepower-chassis /security/keyring # **create certreq**

Step 4 Specify the country code of the country in which the company resides:

Firepower-chassis /security/keyring/certreq* # **set country** *country name*

Step 5 Specify the Domain Name Server (DNS) address associated with the request:

Firepower-chassis /security/keyring/certreq* # **set dns** *DNS Name*

Step 6 Specify the email address associated with the certificate request:

Firepower-chassis /security/keyring/certreq* # **set e-mail** *E-mail name*

Step 7 Specify the IP address of the Firepower 4100/9300 chassis:

Firepower-chassis /security/keyring/certreq* # **set ip** {*certificate request ip-address*|*certificate request ip6-address* }

Step 8 Specify the city or town in which the company requesting the certificate is headquartered:

Firepower-chassis /security/keyring/certreq* # **set locality** *locality name (eg, city)*

Step 9 Specify the organization requesting the certificate:

Firepower-chassis /security/keyring/certreq* # **set org-name** *organization name*

Step 10 Specify the organizational unit:

Firepower-chassis /security/keyring/certreq* # **set org-unit-name** *organizational unit name*

Step 11 Specify an optional password for the certificate request:

Firepower-chassis /security/keyring/certreq* # **set password** *certificate request password*

Step 12 Specify the state or province in which the company requesting the certificate is headquartered:

Firepower-chassis /security/keyring/certreq* # **set state** *state, province or county*

Step 13 Specify the fully qualified domain name of the Firepower 4100/9300 chassis:

Firepower-chassis /security/keyring/certreq* # **set subject-name** *certificate request name*

Step 14 Commit the transaction:

Firepower-chassis /security/keyring/certreq # **commit-buffer**

Step 15 Display the certificate request, which you can copy and send to a trust anchor or certificate authority:

Firepower-chassis /security/keyring # **show certreq**

Example

The following example creates and displays a certificate request with an IPv4 address for a key ring, with advanced options:

```
Firepower-chassis# scope security
Firepower-chassis /security # scope keyring kr220
Firepower-chassis /security/keyring # create certreq
```

```
Firepower-chassis /security/keyring/certreq* # set ip 192.168.200.123
Firepower-chassis /security/keyring/certreq* #  set subject-name sjc04
Firepower-chassis /security/keyring/certreq* # set country US
Firepower-chassis /security/keyring/certreq* # set dns bg1-samc-15A
Firepower-chassis /security/keyring/certreq* # set email test@cisco.com
Firepower-chassis /security/keyring/certreq* # set locality new york city
Firepower-chassis /security/keyring/certreq* # set org-name "Cisco Systems"
Firepower-chassis /security/keyring/certreq* # set org-unit-name Testing
Firepower-chassis /security/keyring/certreq* # set state new york
Firepower-chassis /security/keyring/certreq* # commit-buffer
Firepower-chassis /security/keyring/certreq # show certreq
Certificate request subject name: sjc04
Certificate request ip address: 192.168.200.123
Certificate request e-mail name: test@cisco.com
Certificate request country name: US
State, province or county (full name): New York
Locality name (eg, city): new york city
Organization name (eg, company): Cisco
Organization Unit name (eg, section): Testing
Request:
-----BEGIN CERTIFICATE REQUEST-----
MIIBfTCB5wIBADARMQ8wDQYDVQQDEwZzYW1jMDQwgZ8wDQYJKoZIhvcNAQEBBQAD
gY0AMIGJAoGBALpKn1t8qMZO4UGqILKFXQQc2c8b/vW2rnRF8OPhKbhghLA1YZ1F
JqcYEG5Yl1+vgohLBTd45s0GC8m4RTLJWHo4SwccAUXQ5Zngf45YtX1WsylwUWV4
0re/zgTk/WCd56RfOBvWR2Dtztu2pGA14sd761zLxt29K7R8mzj6CAUVAgMBAAGg
LTArBgkqhkiG9w0BCQ4xHjAcMBoGA1UdEQEB/wQQMA6CBnNhbWMwNIcECsEiXjAN
BgkqhkiG9w0BAQQFAAOBgQCsxN0qUHYGFoQw56RwQueLTNPnrndqUwuZHUO03Teg
nhsyu4satpyiPqVV9viKZ+spvc6x5PWIcTWgHhH8BimOb/0OKuG8kwfIGGsEDlAv
TTYvUP+BZ9OFiPbRIA718S+V8ndXr1HejiQGxlDNqoN+odCXPc5kjoXD0lZTL09H
BA==
-----END CERTIFICATE REQUEST-----

Firepower-chassis /security/keyring/certreq #
```

What to do next

- Copy the text of the certificate request, including the BEGIN and END lines, and save it in a file. Send the file with the certificate request to a trust anchor or certificate authority to obtain a certificate for the key ring.
- Create a trusted point and set the certificate chain for the certificate of trust received from the trust anchor.

Creating a Trusted Point

Procedure

Step 1 Enter security mode:

Firepower-chassis # **scope security**

Step 2 Create a trusted point:

Firepower-chassis /security # **create trustpoint** *name*

Step 3 Specify certificate information for this trusted point:

Firepower-chassis /security/trustpoint # **set certchain** [*certchain*]

If you do not specify certificate information in the command, you are prompted to enter a certificate or a list of trustpoints defining a certification path to the root certificate authority (CA). On the next line following your input, type **ENDOFBUF** to finish.

Important The certificate must be in Base64 encoded X.509 (CER) format.

Step 4 Commit the transaction:

Firepower-chassis /security/trustpoint # **commit-buffer**

Example

The following example creates a trusted point and provides a certificate for the trusted point:

```
Firepower-chassis# scope security
Firepower-chassis /security # create trustpoint tPoint10
Firepower-chassis /security/trustpoint* # set certchain
Enter lines one at a time. Enter ENDOFBUF to finish. Press ^C to abort.
Trustpoint Certificate Chain:
> -----BEGIN CERTIFICATE-----
> MIIDMDCCApmgAwIBAgIBADANBgkqhkiG9w0BAQQFADB0MQswCQYDVQQGEwJVUzEL
> BxMMU2FuIEpvc2UsIENBMRUwEwYDVQQKEwxFeGFtcGxlIEluYy4xEzARBgNVBAsT
> ClRlc3QgR3JvdXAxGTAXBgNVBAMTEHRlc3QuZXhhbXBsZS5jb20xHzAdBgkqhkiG
> 9w0BCQEWEHVzZXJAZXhhbXBsZS5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJ
> AoGBAMZw4nTepNIDhVzb0j7Z2Je4xAG56zmSHRMQeOGHemdh66u2/XAoLx7YCcYU
> ZgAMivyCsKgb/6CjQtsofvtrmC/eAehuK3/SINv7wd6Vv2pBt6ZpXgD4VBNKONDl
> GMbkPayVlQjbG4MD2dx2+H8EH3LMtdZrgKvPxPTE+bF5wZVNAgMBAAGgJTAjBgkq
> hkiG9w0BCQcxFhMUQSBjaGFsbGVuZ2UgcGFzc3dvcmQwDQYJKoZIhvcNAQEFBQAD
> gYEAG61CaJoJaVMhzCl903O6Mg51zq1zXcz75+VFj2I6rH9asckCld3mkOVx5gJU
> Ptt5CVQpNgNLdvbDPSsXretysOhqHmp9+CLv8FDuy1CDYfuaLtvlWvfhevskV0j6
> jtcEMyZ+f7+3yh421ido3nO4MIGeBgNVHSMEgZYwgZOAFLlNjtcEMyZ+f7+3yh42
> 1ido3nO4oXikdjB0MQswCQYDVQQGEwJVUzELMAkGA1UECBMCQ0ExFDASBgNVBAcT
> C1NhbnRhIENsYXJhMRswGQYDVQQKExJOdW92YSBTeXN0ZW1zIEluYy4xFDASBgNV
> BAsTC0VuZ2luZWVyaW5nMQ8wDQYDVQQDEwZ0ZXN0Q0GCAQAwDAYDVR0TBAUwAwEB
> /zANBgkqhkiG9w0BAQQFAAOBgQAhWaRwXNR6B4g6Lsnr+fptHv+WVhB5fKqGQqXc
> wR4pYiO4z42/j9Ijenh75tCKMhW51az8copP1EBmOcyuhf5C6vasrenn1ddkkYt4
> PR0vxGc40whuiozBolesmsmjBbedUCwQgdFDWhDIZJwK5+N3x/kfa2EHU6id1avt
> 4YL5Jg==
> -----END CERTIFICATE-----
> ENDOFBUF
Firepower-chassis /security/trustpoint* # commit-buffer
Firepower-chassis /security/trustpoint #
```

What to do next

Obtain a key ring certificate from the trust anchor or certificate authority and import it into the key ring.

Importing a Certificate into a Key Ring

Before you begin

- Configure a trusted point that contains the certificate chain for the key ring certificate.
- Obtain a key ring certificate from a trust anchor or certificate authority.

Procedure

Step 1 Enter security mode:

Firepower-chassis # **scope security**

Step 2 Enter configuration mode for the key ring that will receive the certificate:

Firepower-chassis /security # **scope keyring** *keyring-name*

Step 3 Specify the trusted point for the trust anchor or certificate authority from which the key ring certificate was obtained:

Firepower-chassis /security/keyring # **set trustpoint** *name*

Step 4 Launch a dialog for entering and uploading the key ring certificate:

Firepower-chassis /security/keyring # **set cert**

At the prompt, paste the certificate text that you received from the trust anchor or certificate authority. On the next line following the certificate, type **ENDOFBUF** to complete the certificate input.

Important The certificate must be in Base64 encoded X.509 (CER) format.

Step 5 Commit the transaction:

Firepower-chassis /security/keyring # **commit-buffer**

Example

The following example specifies the trust point and imports a certificate into a key ring:

```
Firepower-chassis# scope security
Firepower-chassis /security # scope keyring kr220
Firepower-chassis /security/keyring # set trustpoint tPoint10
Firepower-chassis /security/keyring* # set cert
Enter lines one at a time. Enter ENDOFBUF to finish. Press ^C to abort.
Keyring certificate:
> -----BEGIN CERTIFICATE-----
> MIIB/zCCAWgCAQAwgZkxCzAJBgNVBAYTAlVTMQswCQYDVQQIEwJDQTEVMBMGA1UE
> BxMMU2FuIEpvc2UsIENBMRUwEwYDVQQKEwxFeGFtcGxlIEluYy4xEzARBgNVBAsT
> ClRlc3QgR3JvdXAxGTAXBgNVBAMTEHRlc3QuZXhhbXBsZS5jb20xHzAdBgkqhkiG
> 9w0BCQEWEHVzZXJAZXhhbXBsZS5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJ
> AoGBAMZw4nTepNIDhVzb0j7Z2Je4xAG56zmSHRMQeOGHemdh66u2/XAoLx7YCcYU
> ZgAMivyCsKgb/6CjQtsofvtrmC/eAehuK3/SINv7wd6Vv2pBt6ZpXgD4VBNKOND1
> GMbkPayVlQjbG4MD2dx2+H8EH3LMtdZrgKvPxPTE+bF5wZVNAgMBAAGgJTAjBgkq
> hkiG9w0BCQcxFhMUQSBjaGFsbGVuZ2UgcGFzc3dvcmQwDQYJKoZIhvcNAQEFBQAD
> gYEAG61CaJoJaVMhzCl903O6Mg51zq1zXcz75+VFj2I6rH9asckCld3mkOVx5gJU
> Ptt5CVQpNgNLdvbDPSsXretysOhqHmp9+CLv8FDuy1CDYfuaLtvlWvfhevskV0j6
> mK3Ku+YiORnv6DhxrOoqau8r/hyI/L43l7IPN1HhOi3oha4=
> -----END CERTIFICATE-----
> ENDOFBUF
Firepower-chassis /security/keyring* # commit-buffer
Firepower-chassis /security/keyring #
```

What to do next

Configure your HTTPS service with the key ring.

Configuring HTTPS

Caution After you complete the HTTPS configuration, including changing the port and key ring to be used by HTTPS, all current HTTP and HTTPS sessions are closed without warning as soon as you save or commit the transaction.

Procedure

Step 1 Enter system mode:

Firepower-chassis# **scope system**

Step 2 Enter system services mode:

Firepower-chassis /system # **scope services**

Step 3 Enable the HTTPS service:

Firepower-chassis /system/services # **enable https**

Step 4 (Optional) Specify the port to be used for the HTTPS connection:

Firepower-chassis /system/services # **set https port** *port-num*

Step 5 (Optional) Specify the name of the key ring you created for HTTPS:

Firepower-chassis /system/services # **set https keyring** *keyring-name*

Step 6 (Optional) Specify the level of Cipher Suite security used by the domain:

Firepower-chassis /system/services # **set https cipher-suite-mode** *cipher-suite-mode*

cipher-suite-mode can be one of the following keywords:

- **high-strength**
- **medium-strength**
- **low-strength**
- **custom**—Allows you to specify a user-defined Cipher Suite specification string.

Step 7 (Optional) If **cipher-suite-mode** is set to **custom**, specify a custom level of Cipher Suite security for the domain:

Firepower-chassis /system/services # **set https cipher-suite** *cipher-suite-spec-string*

cipher-suite-spec-string can contain up to 256 characters and must conform to the OpenSSL Cipher Suite specifications. You cannot use any spaces or special characters except ! (exclamation point), + (plus sign), - (hyphen), and : (colon). For details, see http://httpd.apache.org/docs/2.0/mod/mod_ssl.html#sslciphersuite.

For example, the medium strength specification string FXOS uses as the default is:
ALL:!ADH:!EXPORT56:!LOW:RC4+RSA:+HIGH:+MEDIUM:+EXP:+eNULL

Note This option is ignored if **cipher-suite-mode** is set to anything other than **custom**.

Step 8 Commit the transaction to the system configuration:

Firepower-chassis /system/services # **commit-buffer**

Example

The following example enables HTTPS, sets the port number to 443, sets the key ring name to kring7984, sets the Cipher Suite security level to high, and commits the transaction:

```
Firepower-chassis# scope system
Firepower-chassis /system # scope services
Firepower-chassis /system/services # enable https
Firepower-chassis /system/services* # set https port 443
Warning: When committed, this closes all the web sessions.
Firepower-chassis /system/services* # set https keyring kring7984
Firepower-chassis /system/services* # set https cipher-suite-mode high
Firepower-chassis /system/services* # commit-buffer
Firepower-chassis /system/services #
```

Changing the HTTPS Port

The HTTPS service is enabled on port 443 by default. You cannot disable HTTPS, but you can change the port to use for HTTPS connections.

Procedure

Step 1 Enter system mode:

Firepower-chassis # **scope system**

Step 2 Enter system services mode:

Firepower-chassis /system # **scope services**

Step 3 Specify the port to use for HTTPS connections:

Firepower-chassis /system/services # **set https port** *port-number*

Specify an integer between 1 and 65535 for *port-number*. HTTPS is enabled on port 443 by default.

Step 4 Commit the transaction to the system configuration:

Firepower /system/services # **commit-buffer**

After changing the HTTPS port, all current HTTPS sessions are closed. Users will need to log back in to the Firepower Chassis Manager using the new port as follows:

https://*<chassis_mgmt_ip_address>*:*<chassis_mgmt_port>*

where <*chassis_mgmt_ip_address*> is the IP address or host name of the Firepower chassis that you entered during initial configuration and <*chassis_mgmt_port*> is the HTTPS port you have just configured.

Example

The following example sets the HTTPS port number to 443 and commits the transaction:

```
Firepower-chassis# scope system
Firepower-chassis /system # scope services
Firepower-chassis /system/services # set https port 444
Warning: When committed, this closes all the web sessions.
Firepower-chassis /system/services* # commit-buffer
Firepower-chassis /system/services #
```

Deleting a Key Ring

Procedure

Step 1 Enter security mode:

Firepower-chassis # **scope security**

Step 2 Delete the named key ring:

Firepower-chassis /security # **delete keyring** *name*

Step 3 Commits the transaction:

Firepower-chassis /security # **commit-buffer**

Example

The following example deletes a key ring:

```
Firepower-chassis# scope security
Firepower-chassis /security # delete keyring key10
Firepower-chassis /security* # commit-buffer
Firepower-chassis /security #
```

Deleting a Trusted Point

Before you begin

Ensure that the trusted point is not used by a key ring.

Procedure

Step 1 Enters security mode:

Firepower-chassis# **scope security**

Step 2 Delete the named trusted point:

Firepower-chassis /security # **delete trustpoint** *name*

Step 3 Commits the transaction:

Firepower-chassis /security # **commit-buffer**

Example

The following example deletes a trusted point:

```
Firepower-chassis# scope security
Firepower-chassis /security # delete trustpoint tPoint10
Firepower-chassis /security* # commit-buffer
Firepower-chassis /security #
```

Disabling HTTPS

Procedure

Step 1 Enter system mode:

Firepower-chassis# **scope system**

Step 2 Enter system services mode:

Firepower-chassis /system # **scope services**

Step 3 Disable the HTTPS service:

Firepower-chassis /system/services # **disable https**

Step 4 Commit the transaction to the system configuration:

Firepower-chassis /system/services # **commit-buffer**

Example

The following example disables HTTPS and commits the transaction:

```
Firepower-chassis# scope system
Firepower-chassis /system # scope services
Firepower-chassis /system/services # disable https
```

```
Firepower-chassis /system/services* # commit-buffer
Firepower-chassis /system/services #
```

Configuring AAA

This section describes authentication, authorization, and accounting. See the following topics for more information:

About AAA

Authentication, Authorization and Accounting (AAA) is a set of services for controlling access to network resources, enforcing policies, assessing usage, and providing the information necessary to bill for services. Authentication identifies the user. Authorization implements policies that determine which resources and services an authenticated user may access. Accounting keeps track of time and data resources that are used for billing and analysis. These processes are considered important for effective network management and security.

Authentication

Authentication provides a way to identify each user, typically by having the user enter a valid user name and valid password before access is granted. The AAA server compares the user's provided credentials with user credentials stored in a database. If the credentials are matched, the user is permitted access to the network. If the credentials do not match, authentication fails and network access is denied.

You can configure the Firepower 4100/9300 chassis to authenticate administrative connections to the chassis, including the following sessions:

- HTTPS
- SSH
- Serial console

Authorization

Authorization is the process of enforcing policies: determining what types of activities, resources, or services each user is permitted to access. After authentication, a user may be authorized for different types of access or activity.

Accounting

Accounting measures the resources a user consumes during access, which may include the amount of system time or the amount of data that a user has sent or received during a session. Accounting is carried out through the logging of session statistics and usage information, which is used for authorization control, billing, trend analysis, resource utilization, and capacity planning activities.

Interaction Between Authentication, Authorization, and Accounting

You can use authentication alone, or with authorization and accounting. Authorization always requires a user to be authenticated first. You can use accounting alone, or with authentication and authorization.

Supported Types of Authentication

FXOS supports the following types of user Authentication:

- **Remote** – The following network AAA services are supported:
  - LDAP
  - RADIUS
  - TACACS+
- **Local** – The Firepower chassis maintains a local database that you can populate with user profiles. You can use this local database instead of AAA servers to provide user authentication, authorization, and accounting.

User Roles

FXOS supports local and remote Authorization in the form of user-role assignment. The roles that can be assigned are:

- **Admin** – Complete read-and-write access to the entire system. The default admin account is assigned this role by default and it cannot be changed.
- **AAA Administrator** – Read-and-write access to users, roles, and AAA configuration. Read access to the rest of the system.
- **Operations** – Read-and-write access to NTP configuration, Smart Call Home configuration for Smart Licensing, and system logs, including syslog servers and faults. Read access to the rest of the system.
- **Read-Only** – Read-only access to system configuration with no privileges to modify the system state.

See User Management, on page 27 for more information about local users and role assignments.

Setting Up AAA

These steps provide a basic outline for setting up Authentication, Authorization and Accounting (AAA) on a Firepower 4100/9300 appliance.

1. Configure the desired type(s) of user authentication:
   - **Local** – User definitions and local authentication are part of User Management, on page 27.
   - **Remote** – Configuring remote AAA server access is part of Platform Settings, specifically:
     - Configuring LDAP Providers, on page 88
     - Configuring RADIUS Providers, on page 92
     - Configuring TACACS+ Providers, on page 95

Note If you will be using remote AAA servers, be sure to enable and configure AAA services on the remote servers before configuring remote AAA server access on the Firepower chassis.

2. Specify the default authentication method—this also is part of User Management, on page 27.

Note If Default Authentication and Console Authentication are both set to use the same remote authentication protocol (RADIUS, TACACS+, or LDAP), you cannot change certain aspects of that server's configuration (for example, deleting that server, or changing its order of assignment) without updating these user settings.

Configuring LDAP Providers

Configuring Properties for LDAP Providers

The properties that you configure in this task are the default settings for all provider connections of this type. If an individual provider includes a setting for any of these properties, the Firepower eXtensible Operating System uses that setting and ignores the default setting.

If you are using Active Directory as your LDAP server, create a user account in the Active Directory server to bind with the Firepower eXtensible Operating System. This account should be given a non-expiring password.

Procedure

Step 1 Enter security mode:

Firepower-chassis# **scope security**

Step 2 Enter security LDAP mode:

Firepower-chassis /security # **scope ldap**

Step 3 Restrict database searches to records that contain the specified attribute:

Firepower-chassis /security/ldap # **set attribute** *attribute*

Step 4 Restrict database searches to records that contain the specified distinguished name:

Firepower-chassis /security/ldap # **set basedn** *distinguished-name*

Step 5 Restrict database searches to records that contain the specified filter:

Firepower-chassis /security/ldap # **set filter** *filter*

where *filter* is the filter attribute to use with your LDAP server, for example *cn=$userid* or *sAMAccountName=$userid*. The filter must include *$userid*.

Step 6 Set the amount of time the system will wait for a response from the LDAP server before noting the server as down:

Firepower-chassis /security/ldap # **set timeout** *seconds*

Step 7 Commit the transaction to the system configuration:

Firepower-chassis /security/ldap # **commit-buffer**

Example

The following example sets the LDAP attribute to CiscoAvPair, the base distinguished name to "DC=cisco-firepower-aaa3,DC=qalab,DC=com", the filter to sAMAccountName=$userid, and the timeout interval to 5 seconds, and commits the transaction:

```
Firepower-chassis# scope security
Firepower-chassis /security # scope ldap
Firepower-chassis /security/ldap # set attribute CiscoAvPair
Firepower-chassis /security/ldap* # set basedn "DC=cisco-firepower-aaa3,DC=qalab,DC=com"
Firepower-chassis /security/ldap* # set filter sAMAccountName=$userid
Firepower-chassis /security/ldap* # set timeout 5
Firepower-chassis /security/ldap* # commit-buffer
Firepower-chassis /security/ldap #
```

Note User login will fail if the DN for an LDAP user exceeds 255 characters.

What to do next

Create an LDAP provider.

Creating an LDAP Provider

Follow these steps to define and configure a LDAP provider—that is, a specific remote server providing LDAP-based AAA services for this Firepower appliance.

Note The Firepower eXtensible Operating System supports a maximum of 16 LDAP providers.

Before you begin

If you are using Active Directory as your LDAP server, create a user account in the Active Directory server to bind with the Firepower eXtensible Operating System. This account should be given a non-expiring password.

Procedure

Step 1 Enter security mode:

Firepower-chassis# **scope security**

Step 2 Enter security LDAP mode:

Firepower-chassis /security # **scope ldap**

Step 3 Create an LDAP server instance and enter security LDAP server mode:

Firepower-chassis /security/ldap # **create server** *server-name*

If SSL is enabled, the *server-name* , typically an IP address or FQDN, must exactly match a Common Name (CN) in the LDAP server's security certificate. Unless an IP address is specified, a DNS server must be configured.

Step 4 (Optional) Set an LDAP attribute that stores the values for the user roles and locales:

Firepower-chassis /security/ldap/server # **set attribute** *attr-name*

This property is always a name-value pair. The system queries the user record for the value that matches this attribute name.

This value is required unless a default attribute has been set for LDAP providers.

Step 5 (Optional) Set the specific distinguished name in the LDAP hierarchy where the server should begin a search when a remote user logs in and the system attempts to get the user's DN based on their user name:

Firepower-chassis /security/ldap/server # **set basedn** *basedn-name*

The length of the base DN can be a maximum of 255 characters minus the length of CN=username, where username identifies the remote user attempting to access Firepower Chassis Manager or the FXOS CLI using LDAP authentication.

This value is required unless a default base DN has been set for LDAP providers.

Step 6 (Optional) Set the distinguished name (DN) for an LDAP database account that has read and search permissions for all objects under the base DN:

Firepower-chassis /security/ldap/server # **set binddn** *binddn-name*

The maximum supported string length is 255 ASCII characters.

Step 7 (Optional) Restrict the LDAP search to user names that match the defined filter.

Firepower-chassis /security/ldap/server # **set filter** *filter-value*

where *filter-value* is the filter attribute to use with your LDAP server; for example *cn=$userid* or *sAMAccountName=$userid*. The filter must include *$userid*.

This value is required unless a default filter has been set for LDAP providers.

Step 8 Specify the password for the LDAP database account specified for Bind DN:

Firepower-chassis /security/ldap/server # **set password**

To set the password, press **Enter** after typing the **set password** command and enter the key value at the prompt.

You can enter any standard ASCII characters except for space, § (section sign), ? (question mark), or = (equal sign).

Step 9 (Optional) Specify the order in which the Firepower eXtensible Operating System uses this provider to authenticate users:

Firepower-chassis /security/ldap/server # **set order** *order-num*

Step 10 (Optional) Specify the port used to communicate with the LDAP server. The standard port number is 389.

Firepower-chassis /security/ldap/server # **set port** *port-num*

Step 11 Enable or disable the use of encryption when communicating with the LDAP server:

Firepower-chassis /security/ldap/server # **set ssl** {**yes** | **no**}

The options are as follows:

- **yes** —Encryption is required. If encryption cannot be negotiated, the connection fails.
- **no** —Encryption is disabled. Authentication information is sent as clear text.

LDAP uses STARTTLS. This allows encrypted communication using port 389.

Step 12 Specify the length of time in seconds the system will spend trying to contact the LDAP database before it times out:

Firepower-chassis /security/ldap/server # **set timeout** *timeout-num*

Enter an integer from 1 to 60 seconds, or enter 0 (zero) to use the global timeout value specified for LDAP providers. The default is 30 seconds.

Step 13 Specify the vendor that is providing the LDAP provider or server details:

Firepower-chassis /security/ldap/server # **set vendor** *{ms-ad | openldap}*

The options are as follows:

- **ms-ad**—LDAP provider is Microsoft Active Directory.
- **openldap**—LDAP provider is not Microsoft Active Directory.

Step 14 Commit the transaction to the system configuration:

Firepower-chassis /security/ldap/server # **commit-buffer**

Example

The following example creates an LDAP server instance named 10.193.169.246, configures the binddn, password, order, port, SSL settings, vendor attribute, and commits the transaction:

```
Firepower-chassis# scope security
Firepower-chassis /security # scope ldap
Firepower-chassis /security/ldap* # create server 10.193.169.246
Firepower-chassis /security/ldap/server* # set binddn
"cn=Administrator,cn=Users,DC=cisco-firepower-aaa3,DC=qalab,DC=com"
Firepower-chassis /security/ldap/server* # set password
Enter the password:
Confirm the password:
Firepower-chassis /security/ldap/server* # set order 2
Firepower-chassis /security/ldap/server* # set port 389
Firepower-chassis /security/ldap/server* # set ssl yes
Firepower-chassis /security/ldap/server* # set timeout 30
Firepower-chassis /security/ldap/server* # set vendor ms-ad
Firepower-chassis /security/ldap/server* # commit-buffer
Firepower-chassis /security/ldap/server #
```

The following example creates an LDAP server instance named 12:31:71:1231:45b1:0011:011:900, configures the binddn, password, order, port, SSL settings, vendor attribute, and commits the transaction:

```
Firepower-chassis# scope security
Firepower-chassis /security # scope ldap
```

```
Firepower-chassis /security/ldap* # create server 12:31:71:1231:45b1:0011:011:900
Firepower-chassis /security/ldap/server* # set binddn
"cn=Administrator,cn=Users,DC=cisco-firepower-aaa3,DC=qalab,DC=com"
Firepower-chassis /security/ldap/server* # set password
Enter the password:
Confirm the password:
Firepower-chassis /security/ldap/server* # set order 1
Firepower-chassis /security/ldap/server* # set port 389
Firepower-chassis /security/ldap/server* # set ssl yes
Firepower-chassis /security/ldap/server* # set timeout 45
Firepower-chassis /security/ldap/server* # set vendor ms-ad
Firepower-chassis /security/ldap/server* # commit-buffer
Firepower-chassis /security/ldap/server #
```

Deleting an LDAP Provider

Procedure

Step 1 Enter security mode:

Firepower-chassis# **scope security**

Step 2 Enter security LDAP mode:

Firepower-chassis /security # **scope ldap**

Step 3 Delete the specified server:

Firepower-chassis /security/ldap # **delete server** *serv-name*

Step 4 Commit the transaction to the system configuration:

Firepower-chassis /security/ldap # **commit-buffer**

Example

The following example deletes the LDAP server called ldap1 and commits the transaction:

```
Firepower-chassis# scope security
Firepower-chassis /security # scope ldap
Firepower-chassis /security/ldap # delete server ldap1
Firepower-chassis /security/ldap* # commit-buffer
Firepower-chassis /security/ldap #
```

Configuring RADIUS Providers

Configuring Properties for RADIUS Providers

The properties that you configure in this task are the default settings for all provider connections of this type. If an individual provider includes a setting for any of these properties, the Firepower eXtensible Operating System uses that setting and ignores this default setting.

Procedure

Step 1 Enter security mode:

Firepower-chassis# **scope security**

Step 2 Enter security RADIUS mode:

Firepower-chassis /security # **scope radius**

Step 3 (Optional) Specify the number of times to retry contacting the RADIUS server before noting the server as down:

Firepower-chassis /security/radius # **set retries** *retry-num*

Step 4 (Optional) Set the amount of time the system will wait for a response from the RADIUS server before noting the server as down:

Firepower-chassis /security/radius # **set timeout** *seconds*

Step 5 Commit the transaction to the system configuration:

Firepower-chassis /security/radius # **commit-buffer**

Example

The following example sets the RADIUS retries to 4, sets the timeout interval to 30 seconds, and commits the transaction:

```
Firepower-chassis# scope security
Firepower-chassis /security # scope radius
Firepower-chassis /security/radius # set retries 4
Firepower-chassis /security/radius* # set timeout 30
Firepower-chassis /security/radius* # commit-buffer
Firepower-chassis /security/radius #
```

What to do next

Create a RADIUS provider.

Creating a RADIUS Provider

Follow these steps to define and configure a RADIUS provider—that is, a specific remote server providing RADIUS-based AAA services for this Firepower appliance.

Note The Firepower eXtensible Operating System supports a maximum of 16 RADIUS providers.

Procedure

Step 1 Enter security mode:

Firepower-chassis# **scope security**

Step 2 Enter security RADIUS mode:

Firepower-chassis /security # **scope radius**

Step 3 Create a RADIUS server instance and enter security RADIUS server mode:

Firepower-chassis /security/radius # **create server** *server-name*

Step 4 (Optional) Specify the port used to communicate with the RADIUS server.

Firepower-chassis /security/radius/server # **set authport** *authport-num*

Step 5 Set the RADIUS server key:

Firepower-chassis /security/radius/server # **set key**

To set the key value, press **Enter** after typing the **set key** command and enter the key value at the prompt.

You can enter any standard ASCII characters except for space, § (section sign), ? (question mark), or = (equal sign).

Step 6 (Optional) Specify when in the order this server will be tried:

Firepower-chassis /security/radius/server # **set order** *order-num*

Step 7 (Optional) Set the number of times to retry communicating with the RADIUS server before noting the server as down:

Firepower-chassis /security/radius/server # **set retries** *retry-num*

Step 8 Specify the length of time in seconds the system will wait for a response from the RADIUS server before noting the server as down:

Firepower-chassis /security/radius/server # **set timeout** *seconds*

Tip It is recommended that you configure a higher **Timeout** value if you select two-factor authentication for RADIUS providers.

Step 9 Commit the transaction to the system configuration:

Firepower-chassis /security/radius/server # **commit-buffer**

Example

The following example creates a server instance named radiusserv7, sets the authentication port to 5858, sets the key to radiuskey321, sets the order to 2, sets the retries to 4, sets the timeout to 30, and commits the transaction:

```
Firepower-chassis# scope security
Firepower-chassis /security # scope radius
Firepower-chassis /security/radius # create server radiusserv7
Firepower-chassis /security/radius/server* # set authport 5858
Firepower-chassis /security/radius/server* # set key
Enter the key: radiuskey321
Confirm the key: radiuskey321
Firepower-chassis /security/radius/server* # set order 2
```

```
Firepower-chassis /security/radius/server* # set retries 4
Firepower-chassis /security/radius/server* # set timeout 30
Firepower-chassis /security/radius/server* # commit-buffer
Firepower-chassis /security/radius/server #
```

Deleting a RADIUS Provider

Procedure

Step 1 Enter security mode:

Firepower-chassis# **scope security**

Step 2 Enter security RADIUS mode:

Firepower-chassis /security # **scope RADIUS**

Step 3 Delete the specified server:

Firepower-chassis /security/radius # **delete server** *serv-name*

Step 4 Commit the transaction to the system configuration:

Firepower-chassis /security/radius # **commit-buffer**

Example

The following example deletes the RADIUS server called radius1 and commits the transaction:

```
Firepower-chassis# scope security
Firepower-chassis /security # scope radius
Firepower-chassis /security/radius # delete server radius1
Firepower-chassis /security/radius* # commit-buffer
Firepower-chassis /security/radius #
```

Configuring TACACS+ Providers

Configuring Properties for TACACS+ Providers

The properties that you configure in this task are default settings for all provider connections of this type. If an individual provider configuration includes a setting for any of these properties, the Firepower eXtensible Operating System uses that setting and ignores this default setting.

Procedure

Step 1 Enter security mode:

Firepower-chassis# **scope security**

Step 2 Enter security TACACS+ mode:

Firepower-chassis /security # **scope tacacs**

Step 3 (Optional) Set the amount of time the system will wait for a response from the TACACS+ server before noting the server as down:

Firepower-chassis /security/tacacs # **set timeout** *seconds*

Enter an integer from 1 to 60 seconds. The default value is 5 seconds.

Step 4 Commit the transaction to the system configuration:

Firepower-chassis /security/tacacs # **commit-buffer**

Example

The following example sets the TACACS+ timeout interval to 45 seconds and commits the transaction:

```
Firepower-chassis# scope security
Firepower-chassis /security # scope tacacs
Firepower-chassis /security/tacacs # set timeout 45
Firepower-chassis /security/tacacs* # commit-buffer
Firepower-chassis /security/tacacs #
```

What to do next

Create a TACACS+ provider.

Creating a TACACS+ Provider

Follow these steps to define and configure a TACACS+ provider—that is, a specific remote server providing TACACS-based AAA services for this Firepower appliance.

Note The Firepower eXtensible Operating System supports a maximum of 16 TACACS+ providers.

Procedure

Step 1 Enter security mode:

Firepower-chassis# **scope security**

Step 2 Enter security TACACS+ mode:

Firepower-chassis /security # **scope tacacs**

Step 3 Create a TACACS+ server instance and enter security TACACS+ server mode:

Firepower-chassis /security/tacacs # **create server** *server-name*

Step 4 Specify the TACACS+ server key:

Firepower-chassis /security/tacacs/server # **set key**

To set the key value, press **Enter** after typing the **set key** command and enter the key value at the prompt.

You can enter any standard ASCII characters except for space, § (section sign), ? (question mark), or = (equal sign).

Step 5 (Optional) Specify when in the order this server will be tried:

Firepower-chassis /security/tacacs/server # **set order** *order-num*

Step 6 Specify the time interval that the system will wait for a response from the TACACS+ server before noting the server as down:

Firepower-chassis /security/tacacs/server # **set timeout** *seconds*

Tip It is recommended that you configure a higher timeout value if you select two-factor authentication for TACACS+ providers.

Step 7 (Optional) Specify the port used to communicate with the TACACS+ server:

Firepower-chassis /security/tacacs/server # **set port** *port-num*

Step 8 Commit the transaction to the system configuration:

Firepower-chassis /security/tacacs/server # **commit-buffer**

Example

The following example creates a server instance named tacacsserv680, sets the key to tacacskey321, sets the order to 4, sets the authentication port to 5859, and commits the transaction:

```
Firepower-chassis# scope security
Firepower-chassis /security # scope tacacs
Firepower-chassis /security/tacacs # create server tacacsserv680
Firepower-chassis /security/tacacs/server* # set key
Enter the key: tacacskey321
Confirm the key: tacacskey321
Firepower-chassis /security/tacacs/server* # set order 4
Firepower-chassis /security/tacacs/server* # set port 5859
Firepower-chassis /security/tacacs/server* # commit-buffer
Firepower-chassis /security/tacacs/server #
```

Deleting a TACACS+ Provider

Procedure

Step 1 Enter security mode:

Firepower-chassis# **scope security**

Step 2 Enter security TACACS+ mode:

Firepower-chassis /security # **scope tacacs**

Step 3 Delete the specified server:

Firepower-chassis /security/tacacs # **delete server** *serv-name*

Step 4 Commit the transaction to the system configuration:

Firepower-chassis /security/tacacs # **commit-buffer**

Example

The following example deletes the TACACS+ server called tacacs1 and commits the transaction:

```
Firepower-chassis# scope security
Firepower-chassis /security # scope tacacs
Firepower-chassis /security/tacacs # delete server tacacs1
Firepower-chassis /security/tacacs* # commit-buffer
Firepower-chassis /security/tacacs #
```

Verifying Remote AAA Server Configurations

The following sections describe how to use the FXOS CLI to determine the current configuration for the various remote AAA servers.

Determining Current FXOS Authentication Configuration

The following example shows you how to use the **show authentication** command to determine the current FXOS authentication settings. In this example, LDAP is the default mode of authentication.

```
firepower# scope security
firepower /security # show authentication
Console authentication: Local
Operational Console authentication: Local
Default authentication: Ldap
Operational Default authentication: Ldap
Role Policy For Remote Users: Assign Default Role
firepower /security #
```

Determining Current LDAP Configuration

The following example shows you how to use the **show server detail** command in ldap mode to determine the current LDAP configuration settings.

```
firepower# scope security
firepower /security # scope ldap
firepower /security/ldap # show server detail

LDAP server:
    Hostname, FQDN or IP address: 10.48.53.132
    Descr:
    Order: 1
    DN to search and read: CN=cisco,CN=Users,DC=fxosldapuser,DC=lab
    Password:
    Port: 389
    SSL: No
    Key:
    Cipher Suite Mode: Medium Strength
    Cipher Suite:
ALL:!DHE-RSK-AES256-CBC-SHA:!EDH-RSA-DES-CBC3-SHA:!EDH-DSS-DES-CBC3-SHA:!DES-CBC3-SHA:!ADH:!3DES:!EXPORT40:!EXPORT56:!LOW:!RC4:!MD5:!IDEA:+HIGH:+MEDIUM:+EXP:+eNULL
```

```
    CRL: Relaxed
    Basedn: CN=Users,DC=fxosldapuser,DC=lab
    User profile attribute: CiscoAVPair
    Filter: cn=$userid
    Timeout: 30
    Ldap Vendor: MS AD
firepower /security/ldap #
```

Determining Current RADIUS Configuration

The following example shows you how to use the **show server detail** command in radius mode to determine the current RADIUS configuration settings.

```
firepower# scope security
firepower /security # scope radius
firepower /security/radius # show server detail

RADIUS server:
    Hostname, FQDN or IP address: 10.48.17.199
    Descr:
    Order: 1
    Auth Port: 1812
    Key: ****
    Timeout: 5
    Retries: 1
firepower /security/radius #
```

Determining Current TACACS+ Configuration

The following example shows you how to use the **show server detail** command in tacacs mode to determine the current TACACS+ configuration settings.

```
firepower# scope security
firepower /security # scope tacacs
firepower /security/tacacs # show server detail

TACACS+ server:
    Hostname, FQDN or IP address: 10.48.17.199
    Descr:
    Order: 1
    Port: 49
    Key: ****
    Timeout: 5
firepower /security/tacacs #
```

Configuring Syslog

System logging is a method of collecting messages from devices to a server running a syslog daemon. Logging to a central syslog server helps in aggregation of logs and alerts. A syslog service accepts messages and stores them in files, or prints them according to a simple configuration file. This form of logging provides protected long-term storage for logs. Logs are useful both in routine troubleshooting and in incident handling.

Procedure

Step 1 Enter monitoring mode:

Firepower-chassis# **scope monitoring**

Step 2 Enable or disable the sending of syslogs to the console:

Firepower-chassis /monitoring # {**enable** | **disable**} **syslog console**

Step 3 (Optional) Select the lowest message level that you want displayed. If syslogs are enabled, the system displays that level and above on the console. The level options are listed in order of decreasing urgency. The default level is Critical.

Firepower-chassis /monitoring # **set syslog console level** {**emergencies** | **alerts** | **critical**}

Step 4 Enable or disable the monitoring of syslog information by the operating system:

Firepower-chassis /monitoring # {**enable** | **disable**} **syslog monitor**

Step 5 (Optional) Select the lowest message level that you want displayed. If the monitor state is enabled, the system displays that level and above. The level options are listed in order of decreasing urgency. The default level is Critical.

Firepower-chassis /monitoring # **set syslog monitor level** {**emergencies** | **alerts** | **critical** | **errors** | **warnings** | **notifications** | **information** | **debugging**}

Note Messages at levels below Critical are displayed on the terminal monitor only if you have entered the **terminal monitor** command.

Step 6 Enable or disable the writing of syslog information to a syslog file:

Firepower-chassis /monitoring # {**enable** | **disable**} **syslog file**

Step 7 Specify the name of the file in which the messages are logged. Up to 16 characters are allowed in the file name.

Firepower-chassis /monitoring # **set syslog file name** *filename*

Step 8 (Optional) Select the lowest message level that you want stored to a file. If the file state is enabled, the system stores that level and above in the syslog file. The level options are listed in order of decreasing urgency. The default level is Critical.

Firepower-chassis /monitoring # **set syslog file level** {**emergencies** | **alerts** | **critical** | **errors** | **warnings** | **notifications** | **information** | **debugging**}

Step 9 (Optional) Specify the maximum file size, in bytes, before the system begins to write over the oldest messages with the newest ones. The range is 4096 to 4194304 bytes.

Firepower-chassis /monitoring # **set syslog file size** *filesize*

Step 10 Configure sending of syslog messages to up to three external syslog servers:

a) Enable or disable the sending of syslog messages to up to three external syslog servers:

Firepower-chassis /monitoring # {**enable** | **disable**} **syslog remote-destination** {**server-1** | **server-2** | **server-3**}

b) (Optional) Select the lowest message level that you want stored to the external log. If the remote-destination is enabled, the system sends that level and above to the external server. The level options are listed in order of decreasing urgency. The default level is Critical.

Firepower-chassis /monitoring # **set syslog remote-destination** {**server-1** | **server-2** | **server-3**} **level** {**emergencies** | **alerts** | **critical** | **errors** | **warnings** | **notifications** | **information** | **debugging**}

c) Specify the hostname or IP address of the specified remote syslog server. Up to 256 characters are allowed in the hostname.

Firepower-chassis /monitoring # **set syslog remote-destination** {**server-1** | **server-2** | **server-3**} **hostname** *hostname*

d) (Optional) Specify the facility level contained in the syslog messages sent to the specified remote syslog server.

Firepower-chassis /monitoring # **set syslog remote-destination** {**server-1** | **server-2** | **server-3**} **facility** {**local0** | **local1** | **local2** | **local3** | **local4** | **local5** | **local6** | **local7**}

Step 11 Configure the local sources. Enter the following command for each of the local sources you want to enable or disable:

Firepower-chassis /monitoring # {**enable** | **disable**} **syslog source** {**audits** | **events** | **faults**}

This can be one of the following:

- **audits**—Enables or disables the logging of all audit log events.
- **events**—Enables or disables the logging of all system events.
- **faults**—Enables or disables the logging of all system faults.

Step 12 Commit the transaction:

Firepower-chassis /monitoring # **commit-buffer**

Example

This example shows how to enable the storage of syslog messages in a local file and commits the transaction:

```
Firepower-chassis# scope monitoring
Firepower-chassis /monitoring # disable syslog console
Firepower-chassis /monitoring* # disable syslog monitor
Firepower-chassis /monitoring* # enable syslog file
Firepower-chassis /monitoring* # set syslog file name SysMsgsFirepower
Firepower-chassis /monitoring* # set syslog file level notifications
Firepower-chassis /monitoring* # set syslog file size 4194304
Firepower-chassis /monitoring* # disable syslog remote-destination server-1
Firepower-chassis /monitoring* # disable syslog remote-destination server-2
Firepower-chassis /monitoring* # disable syslog remote-destination server-3
Firepower-chassis /monitoring* # commit-buffer
Firepower-chassis /monitoring #
```

Configuring DNS Servers

You need to specify a DNS server if the system requires resolution of host names to IP addresses. For example, you cannot use a name such as www.cisco.com when you are configuring a setting on the Firepower chassis if you do not configure a DNS server. You would need to use the IP address of the server, which can be either an IPv4 or an IPv6 address. You can configure up to four DNS servers.

Note When you configure multiple DNS servers, the system searches for the servers only in any random order. If a local management command requires DNS server lookup, it can only search for three DNS servers in random order.

Procedure

Step 1 Enter system mode:

Firepower-chassis # **scope system**

Step 2 Enter system services mode:

Firepower-chassis /system # **scope services**

Step 3 To create or delete a DNS server, enter the appropriate command as follows:

- To configure the system to use a DNS server with the specified IPv4 or IPv6 address:

  Firepower-chassis /system/services # **create dns** *{ip-addr | ip6-addr}*

- To delete a DNS server with the specified IPv4 or IPv6 address:

  Firepower-chassis /system/services # **delete dns** *{ip-addr | ip6-addr}*

Step 4 Commit the transaction to the system configuration:

Firepower /system/services # **commit-buffer**

Example

The following example configures a DNS server with the IPv4 address 192.168.200.105 and commits the transaction:

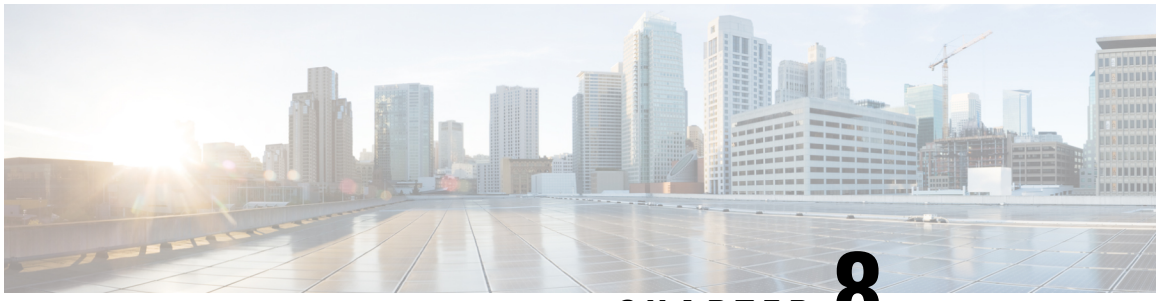
```
Firepower-chassis# scope system
Firepower-chassis /system # scope services
Firepower-chassis /system/services # create dns 192.168.200.105
Firepower-chassis /system/services* # commit-buffer
Firepower-chassis /system/services #
```

The following example configures a DNS server with the IPv6 address 2001:db8::22:F376:FF3B:AB3F and commits the transaction:

```
Firepower-chassis# scope system
Firepower-chassis /system # scope services
Firepower-chassis /system/services # create dns 2001:db8::22:F376:FF3B:AB3F
Firepower-chassis /system/services* # commit-buffer
Firepower-chassis /system/services #
```

The following example deletes the DNS server with the IP address 192.168.200.105 and commits the transaction:

```
Firepower-chassis# scope system
Firepower-chassis /system # scope services
Firepower-chassis /system/services # delete dns 192.168.200.105
Firepower-chassis /system/services* # commit-buffer
Firepower-chassis /system/services #
```



CHAPTER 8

Interface Management

About Firepower Interfaces

The Firepower 4100/9300 chassis supports physical interfaces and EtherChannel (port-channel) interfaces. EtherChannel interfaces can include up to 16 member interfaces of the same type.

Chassis Management Interface

The chassis management interface is used for management of the FXOS Chassis by SSH or Firepower Chassis Manager. This interface is separate from the mgmt-type interface that you assign to the logical devices for application management.

To configure parameters for this interface, you must configure them from the CLI. To view information about this interface in the FXOS CLI, connect to local management and show the management port:

Firepower # **connect local-mgmt**

Firepower(local-mgmt) # **show mgmt-port**

Note that the chassis management interface remains up even if the physical cable or SFP module are unplugged, or if the **mgmt-port shut** command is performed.

Interface Types

Each interface can be one of the following types:

- Data—Use for regular data. Data interfaces cannot be shared between logical devices, and logical devices cannot communicate over the backplane to other logical devices. For traffic on Data interfaces, all traffic must exit the chassis on one interface and return on another interface to reach another logical device.
- Mgmt—Use to manage application instances. These interfaces can be shared by one or more logical devices to access external hosts; logical devices cannot communicate over this interface with other logical

devices that share the interface. You can only assign one management interface per logical device. For ASA: You can later enable management from a data interface; but you must assign a Management interface to the logical device even if you don't intend to use it after you enable data management. For information about the separate chassis management interface, see Chassis Management Interface, on page 105.

- Firepower-eventing—Use as a secondary management interface for FTD devices. To use this interface, you must configure its IP address and other parameters at the FTD CLI. For example, you can separate management traffic from events (such as web events). See the FMC configuration guide for more information. Firepower-eventing interfaces can be shared by one or more logical devices to access external hosts; logical devices cannot communicate over this interface with other logical devices that share the interface.
- Cluster—Use as the cluster control link for a clustered logical device. By default, the cluster control link is automatically created on Port-channel 48. The Cluster type is only supported on EtherChannel interfaces.

FXOS Interfaces vs. Application Interfaces

The Firepower 4100/9300 manages the basic Ethernet settings of physical interfaces and EtherChannel (port-channel) interfaces. Within the application, you configure higher level settings. For example, you can only create EtherChannels in FXOS; but you can assign an IP address to the EtherChannel within the application.

The following sections describe the interaction between FXOS and the application for interfaces.

VLAN Subinterfaces

For all logical devices, you can create VLAN subinterfaces within the application.

Independent Interface States in the Chassis and in the Application

You can administratively enable and disable interfaces in both the chassis and in the application. For an interface to be operational, the interface must be enabled in both operating systems. Because the interface state is controlled independently, you may have a mismatch between the chassis and application.

Jumbo Frame Support

The Firepower 4100/9300 chassis has support for jumbo frames enabled by default. To enable jumbo frame support on a specific logical device installed on the Firepower 4100/9300 chassis, you will need to configure the appropriate MTU settings for the interfaces on the logical device.

The maximum MTU that is supported for the application on the Firepower 4100/9300 chassis is 9000.

Guidelines and Limitations for Firepower Interfaces

Inline Sets for FTD

Default MAC Addresses

Default MAC address assignments depend on the type of interface.

- Physical interfaces—The physical interface uses the burned-in MAC address.
- EtherChannels—For an EtherChannel, all interfaces that are part of the channel group share the same MAC address. This feature makes the EtherChannel transparent to network applications and users, because they only see the one logical connection; they have no knowledge of the individual links. The port-channel interface uses a unique MAC address from a pool; interface membership does not affect the MAC address.

Configure Interfaces

By default, physical interfaces are disabled. You can enable interfaces, add EtherChannels, and edit interface properties.

Configure a Physical Interface

You can physically enable and disable interfaces, as well as set the interface speed and duplex. To use an interface, it must be physically enabled in FXOS and logically enabled in the application.

Before you begin

- Interfaces that are already a member of an EtherChannel cannot be modified individually. Be sure to configure settings before you add it to the EtherChannel.

Procedure

Step 1 Enter interface mode.

scope eth-uplink

scope fabric a

Step 2 Enable the interface.

enter interface *interface_id*

enable

Example:

```
Firepower /eth-uplink/fabric # enter interface Ethernet1/8
Firepower /eth-uplink/fabric/interface # enable
```

Note Interfaces that are already a member of a port-channel cannot be modified individually. If you use the **enter interface** or **scope interface** command on an interface that is a member of a port channel, you will receive an error stating that the object does not exist. You should edit interfaces using the **enter interface** command before you add them to a port-channel.

Step 3 (Optional) Set the interface type.

set port-type {**data** | **mgmt** | **firepower-eventing** | **cluster**}

Example:

```
Firepower /eth-uplink/fabric/interface # set port-type mgmt
```

The **data** keyword is the default type. Do not choose the **cluster** keyword; by default, the cluster control link is automatically created on Port-channel 48.

Step 4 Enable or disable autonegotiation, if supported for your interface.

set auto-negotiation {**on** | **off**}

Example:

```
Firepower /eth-uplink/fabric/interface* # set auto-negotiation off
```

Step 5 Set the interface speed.

set admin-speed {**10mbps** | **100mbps** | **1gbps** | **10gbps** | **40gbps** | **100gbps**}

Example:

```
Firepower /eth-uplink/fabric/interface* # set admin-speed 1gbps
```

Step 6 Set the interface duplex mode.

set admin-duplex {**fullduplex** | **halfduplex**}

Example:

```
Firepower /eth-uplink/fabric/interface* # set admin-duplex halfduplex
```

Step 7 If you edited the default flow control policy, it is already applied to interfaces. If you created a new policy, apply it to the interface.

set flow-control-policy *name*

Example:

```
Firepower /eth-uplink/fabric/interface* # set flow-control-policy flow1
```

Step 8 Save the configuration.

commit-buffer

Example:

```
Firepower /eth-uplink/fabric/interface* # commit-buffer
Firepower /eth-uplink/fabric/interface #
```

Add an EtherChannel (Port Channel)

An EtherChannel (also known as a port channel) can include up to 16 member interfaces of the same media type and capacity, and must be set to the same speed and duplex. The media type can be either RJ-45 or SFP; SFPs of different types (copper and fiber) can be mixed. You cannot mix interface capacities (for example 1GB and 10GB interfaces) by setting the speed to be lower on the larger-capacity interface. The Link Aggregation Control Protocol (LACP) aggregates interfaces by exchanging the Link Aggregation Control Protocol Data Units (LACPDUs) between two network devices.

LACP coordinates the automatic addition and deletion of links to the EtherChannel without user intervention. It also handles misconfigurations and checks that both ends of member interfaces are connected to the correct channel group.

When the Firepower 4100/9300 chassis creates an EtherChannel, the EtherChannel stays in a **Suspended** state until you assign it to a logical device, even if the physical link is up. The EtherChannel will be brought out of this **Suspended** state in the following situations:

- The EtherChannel is added as a data or management interface for a standalone logical device
- The EtherChannel is added as a management interface or cluster control link for a logical device that is part of a cluster
- The EtherChannel is added as a data interface for a logical device that is part of a cluster and at least one unit has joined the cluster

Note that the EtherChannel does not come up until you assign it to a logical device. If the EtherChannel is removed from the logical device or the logical device is deleted, the EtherChannel will revert to a **Suspended** state.

Procedure

Step 1 Enter interface mode:

scope eth-uplink

scope fabric a

Step 2 Create the port-channel:

create port-channel *id*

enable

Step 3 Assign member interfaces:

create member-port *interface_id*

Example:

```
Firepower /eth-uplink/fabric/port-channel* # create member-port Ethernet1/1
Firepower /eth-uplink/fabric/port-channel/member-port* # exit
Firepower /eth-uplink/fabric/port-channel* # create member-port Ethernet1/2
Firepower /eth-uplink/fabric/port-channel/member-port* # exit
Firepower /eth-uplink/fabric/port-channel* # create member-port Ethernet1/3
Firepower /eth-uplink/fabric/port-channel/member-port* # exit
Firepower /eth-uplink/fabric/port-channel* # create member-port Ethernet1/4
```

```
Firepower /eth-uplink/fabric/port-channel/member-port* # exit
```

Step 4 (Optional) Set the interface type.

set port-type {**data** | **mgmt** | **firepower-eventing** | **cluster**}

Example:

```
Firepower /eth-uplink/fabric/port-channel # set port-type data
```

The **data** keyword is the default type. Do not choose the **cluster** keyword unless you want to use this port-channel as the cluster control link instead of the default.

Step 5 (Optional) Set the interface speed for all members of the port-channel.

set speed {**10mbps** | **100mbps** | **1gbps** | **10gbps** | **40gbps** | **100gbps**}

Example:

```
Firepower /eth-uplink/fabric/port-channel* # set speed 1gbps
```

Step 6 (Optional) Set the duplex for all members of the port-channel.

set duplex {**fullduplex** | **halfduplex**}

Example:

```
Firepower /eth-uplink/fabric/port-channel* # set duplex fullduplex
```

Step 7 Enable or disable autonegotiation, if supported for your interface.

set auto-negotiation {**on** | **off**}

Example:

```
Firepower /eth-uplink/fabric/interface* # set auto-negotiation off
```

Step 8 If you edited the default flow control policy, it is already applied to interfaces. If you created a new policy, apply it to the interface.

set flow-control-policy *name*

Example:

```
Firepower /eth-uplink/fabric/interface* # set flow-control-policy flow1
```

Step 9 Commit the configuration:

commit-buffer

Configure Breakout Cables

The following procedure shows how to configure breakout cables for use with the Firepower 4100/9300 chassis. You can use a breakout cable to provide four 10 Gbps ports in place of a single 40 Gbps port.

Procedure

Step 1 To create a new breakout, use the following commands:

a) Enter cabling mode:

scope cabling

scope fabric a

b) Create the breakout:

create breakout *network_module_slot port*

Example:

```
Firepower /cabling/fabric/ # create breakout 2 1
```

c) Commit the configuration:

commit-buffer

This will cause an automatic reboot. If you are configuring more than one breakout, you should create all of them before you issue the commit-buffer command.

Step 2 To enable/configure the breakout ports, use the following commands:

a) Enter interface mode:

scope eth-uplink

scope fabric a

scope aggr-interface *network_module_slot port*

Note Interfaces that are already a member of a port-channel cannot be modified individually. If you use the **enter interface** or **scope interface** command on an interface that is a member of a port channel, you will receive an error stating that the object does not exist. You should edit interfaces using the **enter interface** command before you add them to a port-channel.

b) Use the **set** command to configure the interface speed and port type.

Use the **enable** or **disable** command to set the administrative state of the interface.

c) Commit the configuration:

commit-buffer

Configure a Flow Control Policy

Flow control policies determine whether the Ethernet ports send and receive IEEE 802.3x pause frames when the receive buffer for a port fills. These pause frames request that the transmitting port stop sending data for a few milliseconds until the buffer clears. For flow control to work between devices, you must enable the corresponding receive and send flow control parameters for both devices.

The default policy disables send and receive control, and sets the priority to autonegotiate.

Procedure

Step 1 Enter eth-uplink and then flow-control mode.

scope eth-uplink

scope flow-control

Example:

```
firepower-4110# scope eth-uplink
firepower-4110 /eth-uplink # scope flow-control
firepower-4110 /eth-uplink/flow-control #
```

Step 2 Edit or create a flow control policy.

enter policy *name*

If you want to edit the default policy, enter **default** for the name.

Example:

```
firepower-4110 /eth-uplink/flow-control # enter policy default
firepower-4110 /eth-uplink/flow-control/policy* #
```

Step 3 Set the priority.

set prio {**auto** | **on**}

The priority sets whether to negotiate or enable PPP for this link.

Example:

```
firepower-4110 /eth-uplink/flow-control/policy* # set prio on
```

Step 4 Enable or disable flow control receive pauses.

set receive {**on** | **off**}

- **on**—Pause requests are honored and all traffic is halted on that uplink port until the network cancels the pause request.
- **off**—Pause requests from the network are ignored and traffic flow continues as normal.

Example:

```
firepower-4110 /eth-uplink/flow-control/policy* # set receive on
```

Step 5 Enable or disable flow control send pauses.

set send {**on** | **off**}

- **on**—The Firepower 4100/9300 sends a pause request to the network if the incoming packet rate becomes too high. The pause remains in effect for a few milliseconds before traffic is reset to normal levels.
- **off**—Traffic on the port flows normally regardless of the packet load.

Example:

```
firepower-4110 /eth-uplink/flow-control/policy* # set send on
```

Step 6 Save the configuration.

commit-buffer

Example:

```
firepower-4110 /eth-uplink/flow-control/policy* # commit-buffer
firepower-4110 /eth-uplink/flow-control/policy #
```

Example

The following example configures a flow control policy.

```
firepower-4110# scope eth-uplink
firepower-4110 /eth-uplink # scope flow-control
firepower-4110 /eth-uplink/flow-control # enter policy FlowControlPolicy23
firepower-4110 /eth-uplink/flow-control/policy* # set prio auto
firepower-4110 /eth-uplink/flow-control/policy* # set receive on
firepower-4110 /eth-uplink/flow-control/policy* # set send on
firepower-4110 /eth-uplink/flow-control/policy* # commit-buffer
firepower-4110 /eth-uplink/flow-control/policy #
```

Monitoring Interfaces

- **show interface**

  Shows interface status.

  Note Interfaces that act as ports in port channels do not appear in this list.

```
Firepower# scope eth-uplink
```

```
Firepower /eth-uplink # scope fabric a
Firepower /eth-uplink/fabric # show interface

Interface:
    Port Name       Port Type          Admin State Oper State       State Reason
    --------------- ------------------ ----------- ---------------- ------------
    Ethernet1/1     Mgmt               Enabled     Up
    Ethernet1/2     Data               Enabled     Link Down        Link failure or
not-connected
    Ethernet1/3     Data               Enabled     Up
    Ethernet1/4     Data               Enabled     Sfp Not Present  Unknown
    Ethernet1/6     Data               Enabled     Sfp Not Present  Unknown
    Ethernet1/7     Data               Enabled     Sfp Not Present  Unknown
    Ethernet1/8     Data               Disabled    Sfp Not Present  Unknown
    Ethernet2/1     Data               Enabled     Up
    Ethernet2/2     Data               Enabled     Up
    Ethernet2/4     Data               Enabled     Up
    Ethernet2/5     Data               Enabled     Up
    Ethernet2/6     Data               Enabled     Up
    Ethernet3/2     Data               Enabled     Up
    Ethernet3/4     Data               Enabled     Up
```

- **show port-channel**

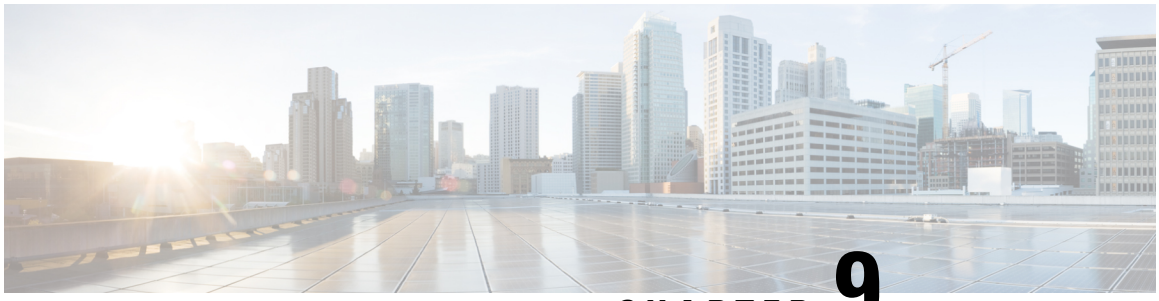
  Shows port-channel status.

```
Firepower# scope eth-uplink
Firepower /eth-uplink # scope fabric a
Firepower /eth-uplink/fabric # show port-channel

Port Channel:
    Port Channel Id Name             Port Type          Admin State Oper State
Port Channel Mode State Reason
    --------------- ---------------- ------------------ ----------- ----------------
---------------- ------------
    1               Port-channel1    Data               Enabled     Up
Active
    2               Port-channel2    Data               Enabled     Failed
Active            No operational members
    48              Port-channel48   Cluster            Enabled     Up
Active
```

History for Interfaces

Feature Name	Platform Releases	Feature Information
Firepower-eventing type interface for FTD	1.1.4	You can specify an interface as firepower-eventing for use with the FTD. This interface is a secondary management interface for FTD devices. To use this interface, you must configure its IP address and other parameters at the FTD CLI. For example, you can separate management traffic from events (such as web events). See the "Management Interfaces" section in the Firepower Management Center configuration guide *System Configuration* chapter. New/Modified FXOS commands: **set port-type firepower-eventing, show interface**



CHAPTER 9

Logical Devices

About Logical Devices

A logical device lets you run one application instance (either ASA or Firepower Threat Defense) and also one optional decorator application (Radware DefensePro) to form a service chain .

When you add a logical device, you also define the application instance type and version, assign interfaces, and configure bootstrap settings that are pushed to the application configuration.

Note For the Firepower 9300, you must install the same application instance type (ASA or FTD) on all modules in the chassis; different types are not supported at this time. Note that modules can run different versions of an application instance type.

Standalone and Clustered Logical Devices

You can add the following logical device types:

- Standalone—A standalone logical device operates as a standalone unit or as a unit in a High Availability pair.
- Cluster—A clustered logical device lets you group multiple units together, providing all the convenience of a single device (management, integration into a network) while achieving the increased throughput

and redundancy of multiple devices. Multiple module devices, like the Firepower 9300, support intra-chassis clustering. For the Firepower 9300, all three modules must participate in the cluster.

Requirements and Prerequisites for Logical Devices

See the following sections for requirements and prerequisites.

Requirements and Prerequisites for Hardware and Software Combinations

The Firepower 4100/9300 supports multiple models, security modules, application types, and high availability and scalability features. See the following requirements for allowed combinations.

Firepower 9300 Requirements

The Firepower 9300 includes 3 security module slots and multiple types of security modules. See the following requirements:

- Security Module Types—
- Clustering—All security modules in the cluster, whether it is intra-chassis or inter-chassis, must be the same type. You can have different quantities of installed security modules in each chassis, although all modules present in the chassis must belong to the cluster including any empty slots. For example, you can install 2 SM-36s in chassis 1, and 3 SM-36s in chassis 2.
- High Availability—High Availability is only supported between same-type modules on the Firepower 9300.
- ASA and FTD application types—
- ASA or FTD versions—You can run different versions of an application instance type on separate modules. For example, you can install FTD 6.3 on module 1, FTD 6.4 on module 2, and FTD 6.5 on module 3.

Firepower 4100 Requirements

The Firepower 4100 comes in multiple models. See the following requirements:

- Clustering—All chassis in the cluster must be the same model.
- High Availability—High Availability is only supported between same-type models.
- ASA and FTD application types—The Firepower 4100 can only run a single application type.

Requirements and Prerequisites for Clustering

Cluster Model Support

- ASA on the Firepower 9300—Maximum 16 modules. For example, you can use 1 module in 16 chassis, or 2 modules in 8 chassis, or any combination that provides a maximum of 16 modules. Note that all modules in a chassis must belong to the cluster. Supported for intra-chassis, inter-chassis, and inter-site clustering.

- ASA on the Firepower 4100 series—Maximum 6 chassis. Supported for inter-chassis and inter-site clustering.
- FTD on the Firepower 9300—Maximum 3 modules in 1 chassis. Supported for intra-chassis clustering.
- Radware DefensePro—Supported for intra-chassis clustering with the ASA.

Clustering Hardware and Software Requirements

All chassis in a cluster:

- For the Firepower 4100 series: All chassis must be the same model. For the Firepower 9300: All security modules must be the same type. For example, if you use clustering, all modules in the Firepower 9300 must be SM-40s. You can have different quantities of installed security modules in each chassis, although all modules present in the chassis must belong to the cluster including any empty slots.
- Must run the identical FXOS software except at the time of an image upgrade.
- Must include the same interface configuration for interfaces you assign to the cluster, such as the same Management interface, EtherChannels, active interfaces, speed and duplex, and so on. You can use different network module types on the chassis as long as the capacity matches for the same interface IDs and interfaces can successfully bundle in the same spanned EtherChannel. Note that all data interfaces must be EtherChannels in inter-chassis clustering. If you change the interfaces in FXOS after you enable clustering (by adding or removing interface modules, or configuring EtherChannels, for example), then perform the same changes on each chassis, starting with the data units, and ending with the control unit.
- Must use the same NTP server. Do not set the time manually.
- ASA: Each FXOS chassis must be registered with the License Authority or satellite server. There is no extra cost for data units. For Firepower Threat Defense, all licensing is handled by the Firepower Management Center.

Switch Requirements for Inter-Chassis Clustering

- Be sure to complete the switch configuration and successfully connect all the EtherChannels from the chassis to the switch(es) before you configure clustering on the Firepower 4100/9300 chassis.
- For supported switch characteristics, see Cisco FXOS Compatibility.

Sizing the Data Center Interconnect for Inter-Site Clustering

You should reserve bandwidth on the data center interconnect (DCI) for cluster control link traffic equivalent to the following calculation:

$$\frac{\text{\# of cluster members per site}}{2} \times \text{cluster control link size per member}$$

If the number of members differs at each site, use the larger number for your calculation. The minimum bandwidth for the DCI should not be less than the size of the cluster control link for one member.

For example:

- For 4 members at 2 sites:
  - 4 cluster members total

- 2 members at each site
- 5 Gbps cluster control link per member

Reserved DCI bandwidth = 5 Gbps (2/2 x 5 Gbps).

- For 6 members at 3 sites, the size increases:
  - 6 cluster members total
  - 3 members at site 1, 2 members at site 2, and 1 member at site 3
  - 10 Gbps cluster control link per member

  Reserved DCI bandwidth = 15 Gbps (3/2 x 10 Gbps).
- For 2 members at 2 sites:
  - 2 cluster members total
  - 1 member at each site
  - 10 Gbps cluster control link per member

  Reserved DCI bandwidth = 10 Gbps (1/2 x 10 Gbps = 5 Gbps; but the minimum bandwidth should not be less than the size of the cluster control link (10 Gbps)).

Requirements and Prerequisites for High Availability

- The two units in a High Availability Failover configuration must:
  - Be on a separate chassis; intra-chassis High Availability for the Firepower 9300 is not supported.
  - Be the same model.
  - Have the same interfaces assigned to the High Availability logical devices.
  - Have the same number and types of interfaces. All interfaces must be preconfigured in FXOS identically before you enable High Availability.
- For High Availability system requirements, see.

Guidelines and Limitations for Logical Devices

See the following sections for guidelines and limitations.

General Guidelines and Limitations

Firewall Mode

You can set the firewall mode to routed or transparent in the bootstrap configuration for the FTD.

High Availability

- Configure high availability within the application configuration.
- You can use any data interfaces as the failover and state links.

Context Mode

- Multiple context mode is only supported on the ASA.

Clustering Guidelines and Limitations

Switches for Inter-Chassis Clustering

- For the ASR 9006, if you want to set a non-default MTU, set the ASR interface MTU to be 14 bytes higher than the cluster device MTU. Otherwise, OSPF adjacency peering attempts may fail unless the **mtu-ignore** option is used. Note that the cluster device MTU should match the ASR *IPv4* MTU.
- On the switch(es) for the cluster control link interfaces, you can optionally enable Spanning Tree PortFast on the switch ports connected to the cluster unit to speed up the join process for new units.
- On the switch, we recommend that you use one of the following EtherChannel load-balancing algorithms: **source-dest-ip** or **source-dest-ip-port** (see the Cisco Nexus OS and Cisco IOS **port-channel load-balance** command). Do not use a **vlan** keyword in the load-balance algorithm because it can cause unevenly distributed traffic to the devices in a cluster.
- If you change the load-balancing algorithm of the EtherChannel on the switch, the EtherChannel interface on the switch temporarily stops forwarding traffic, and the Spanning Tree Protocol restarts. There will be a delay before traffic starts flowing again.
- Some switches do not support dynamic port priority with LACP (active and standby links). You can disable dynamic port priority to provide better compatibility with Spanned EtherChannels.
- Switches on the cluster control link path should not verify the L4 checksum. Redirected traffic over the cluster control link does not have a correct L4 checksum. Switches that verify the L4 checksum could cause traffic to be dropped.
- Port-channel bundling downtime should not exceed the configured keepalive interval.
- On Supervisor 2T EtherChannels, the default hash distribution algorithm is adaptive. To avoid asymmetric traffic in a VSS design, change the hash algorithm on the port-channel connected to the cluster device to fixed:

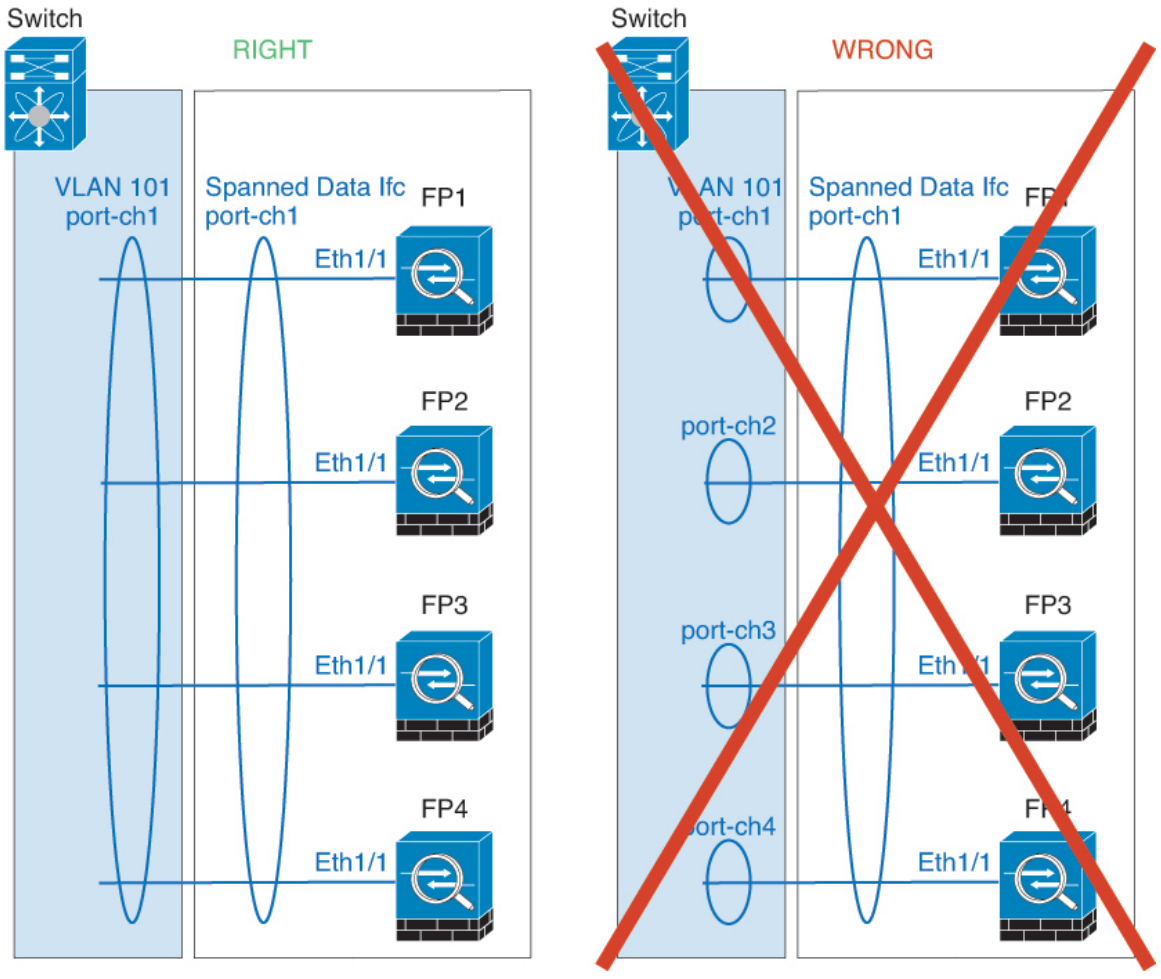
  router(config)# **port-channel** *id* **hash-distribution fixed**

  Do not change the algorithm globally; you may want to take advantage of the adaptive algorithm for the VSS peer link.

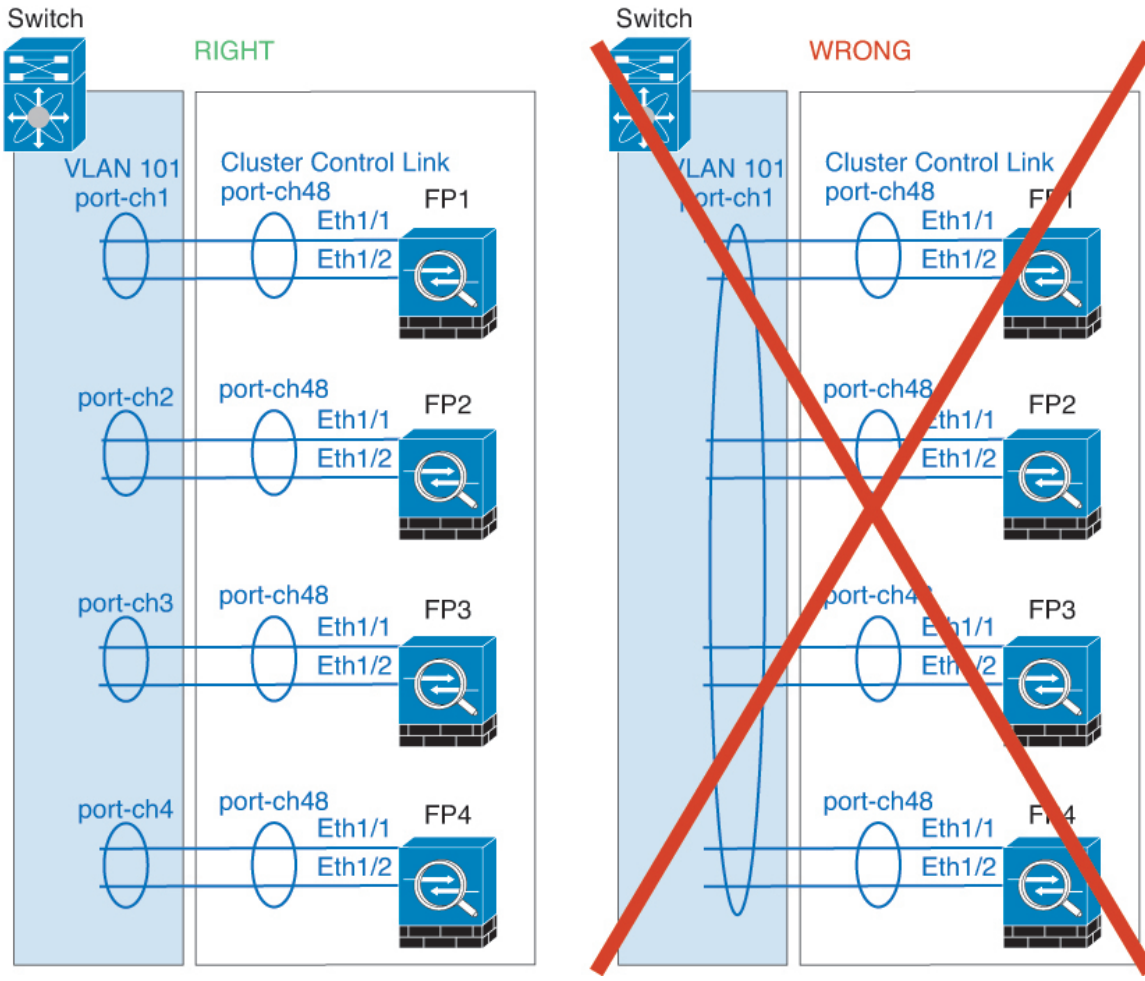
- Firepower 4100/9300 clusters support LACP graceful convergence. So you can leave LACP graceful convergence enabled on connected Cisco Nexus switches.
- When you see slow bundling of a Spanned EtherChannel on the switch, you can enable LACP rate fast for an individual interface on the switch. FXOS EtherChannels have the LACP rate set to fast by default. Note that some switches, such as the Nexus series, do not support LACP rate fast when performing in-service software upgrades (ISSUs), so we do not recommend using ISSUs with clustering.

EtherChannels for Inter-Chassis Clustering

- In Catalyst 3750-X Cisco IOS software versions earlier than 15.1(1)S2, the cluster unit did not support connecting an EtherChannel to a switch stack. With default switch settings, if the cluster unit EtherChannel is connected cross stack, and if the control unit switch is powered down, then the EtherChannel connected to the remaining switch will not come up. To improve compatibility, set the **stack-mac persistent timer** command to a large enough value to account for reload time; for example, 8 minutes or 0 for indefinite. Or, you can upgrade to more a more stable switch software version, such as 15.1(1)S2.
- Spanned vs. Device-Local EtherChannel Configuration—Be sure to configure the switch appropriately for Spanned EtherChannels vs. Device-local EtherChannels.
  - Spanned EtherChannels—For cluster unit *Spanned* EtherChannels, which span across all members of the cluster, the interfaces are combined into a single EtherChannel on the switch. Make sure each interface is in the same channel group on the switch.



  - Device-local EtherChannels—For cluster unit *Device-local* EtherChannels including any EtherChannels configured for the cluster control link, be sure to configure discrete EtherChannels on the switch; do not combine multiple cluster unit EtherChannels into one EtherChannel on the switch.



Inter-Site Clustering

See the following guidelines for inter-site clustering:

- The cluster control link latency must be less than 20 ms round-trip time (RTT).
- The cluster control link must be reliable, with no out-of-order or dropped packets; for example, you should use a dedicated link.
- Do not configure connection rebalancing; you do not want connections rebalanced to cluster members at a different site.
- The cluster implementation does not differentiate between members at multiple sites for incoming connections; therefore, connection roles for a given connection may span across sites. This is expected behavior.
- For transparent mode, if the cluster is placed between a pair of inside and outside routers (AKA North-South insertion), you must ensure that both inside routers share a MAC address, and also that both outside routers share a MAC address. When a cluster member at site 1 forwards a connection to a member at site 2, the destination MAC address is preserved. The packet will only reach the router at site 2 if the MAC address is the same as the router at site 1.

- For transparent mode, if the cluster is placed between data networks and the gateway router at each site for firewalling between internal networks (AKA East-West insertion), then each gateway router should use a First Hop Redundancy Protocol (FHRP) such as HSRP to provide identical virtual IP and MAC address destinations at each site. The data VLANs are extended across the sites using Overlay Transport Virtualization (OTV), or something similar. You need to create filters to prevent traffic that is destined to the local gateway router from being sent over the DCI to the other site. If the gateway router becomes unreachable at one site, you need to remove any filters so traffic can successfully reach the other site's gateway.
- For routed mode using Spanned EtherChannel, configure site-specific MAC addresses. Extend the data VLANs across the sites using OTV, or something similar. You need to create filters to prevent traffic that is destined to the global MAC address from being sent over the DCI to the other site. If the cluster becomes unreachable at one site, you need to remove any filters so traffic can successfully reach the other site's cluster units. Dynamic routing is not supported when an inter-site cluster acts as the first hop router for an extended segment.

Additional Guidelines

- When adding a unit to an existing cluster, or when reloading a unit, there will be a temporary, limited packet/connection drop; this is expected behavior. In some cases, the dropped packets can hang connections; for example, dropping a FIN/ACK packet for an FTP connection will make the FTP client hang. In this case, you need to reestablish the FTP connection.
- If you use a Windows 2003 server connected to a Spanned EtherChannel interface, when the syslog server port is down, and the server does not throttle ICMP error messages, then large numbers of ICMP messages are sent back to the cluster. These messages can result in some units of the cluster experiencing high CPU, which can affect performance. We recommend that you throttle ICMP error messages.
- We recommend connecting EtherChannels to a VSS or vPC for redundancy.
- Within a chassis, you cannot cluster some security modules and run other security modules in standalone mode; you must include all security modules in the cluster.

Defaults

- The cluster health check feature is enabled by default with the holdtime of 3 seconds. Interface health monitoring is enabled on all interfaces by default.
- The cluster auto-rejoin feature for a failed cluster control link is set to unlimited attempts every 5 minutes.
- The cluster auto-rejoin feature for a failed data interface is set to 3 attempts every 5 minutes, with the increasing interval set to 2.
- Connection replication delay of 5 seconds is enabled by default for HTTP traffic.

Add a Standalone Logical Device

Standalone logical devices can be used alone or as high availability units. For more information about high availability usage, see Add a High Availability Pair, on page 136.

Add a Standalone ASA

Standalone logical devices work either alone or in a High Availability pair. On the Firepower 9300 with multiple security modules, you can deploy either a cluster or standalone devices. The cluster must use all modules, so you cannot mix and match a 2-module cluster plus a single standalone device, for example.

You can deploy a routed firewall mode ASA from the Firepower 4100/9300 chassis.

For multiple context mode, you must first deploy the logical device, and then enable multiple context mode in the ASA application.

Before you begin

- Download the application image you want to use for the logical device from Cisco.com, and then download that image to the Firepower 4100/9300 chassis.

Note For the Firepower 9300, you must install the same application instance type (ASA or FTD) on all modules in the chassis; different types are not supported at this time. Note that modules can run different versions of an application instance type.

- Configure a management interface to use with the logical device. The management interface is required. Note that this management interface is not the same as the chassis management port that is used only for chassis management (in FXOS, you might see it displayed as MGMT, management0, or other similar names).
- Gather the following information:
  - Interface IDs for this device
  - Management interface IP address and network mask
  - Gateway IP address

Procedure

Step 1 Enter security services mode.

scope ssa

Example:

```
Firepower# scope ssa
Firepower /ssa #
```

Step 2 Set the application instance image version.

a) View available images. Note the Version number that you want to use.

show app

Example:

```
Firepower /ssa # show app
    Name       Version         Author     Supported Deploy Types CSP Type    Is Default
 App
    ---------- --------------- ---------- ---------------------- -----------
--------------
    asa        9.9.1           cisco      Native                 Application No
    asa        9.10.1          cisco      Native                 Application Yes
    ftd        6.2.3           cisco      Native                 Application Yes
```

b) Set the scope to the security module/engine slot.

scope slot *slot_id*

The *slot_id* is always 1 for the Firepower 4100, and 1, 2, or 3 for the Firepower 9300.

Example:

```
Firepower /ssa # scope slot 1
Firepower /ssa/slot #
```

c) Create the application instance.

enter app-instance asa

Example:

```
Firepower /ssa/slot # enter app-instance asa
Firepower /ssa/slot/app-instance* #
```

d) Set the ASA image version.

set startup-version *version*

Example:

```
Firepower /ssa/slot/app-instance* # set startup-version 9.10.1
```

e) Exit to slot mode.

exit

Example:

```
Firepower /ssa/slot/app-instance* # exit
Firepower /ssa/slot* #
```

f) Exit to ssa mode.

exit

Example:

```
Firepower /ssa/slot* # exit
Firepower /ssa* #
```

Example:

```
Firepower /ssa # scope slot 1
Firepower /ssa/slot # enter app-instance asa
Firepower /ssa/slot/app-instance* # set startup-version 9.10.1
Firepower /ssa/slot/app-instance* # exit
Firepower /ssa/slot* # exit
Firepower /ssa* #
```

Step 3 Create the logical device.

enter logical-device *device_name* **asa** *slot_id* **standalone**

Example:

```
Firepower /ssa # enter logical-device ASA1 asa 1 standalone
Firepower /ssa/logical-device* #
```

Step 4 Assign the management and data interfaces to the logical device. Repeat for each interface.

create external-port-link *name interface_id* **asa**

set description *description*

exit

- *name*—The name is used by the Firepower 4100/9300 chassis supervisor; it is not the interface name used in the ASA configuration.
- *description*—Use quotes (") around phrases with spaces.

The management interface is not the same as the chassis management port. You will later enable and configure the data interfaces on the ASA, including setting the IP addresses.

Example:

```
Firepower /ssa/logical-device* # create external-port-link inside Ethernet1/1 asa
Firepower /ssa/logical-device/external-port-link* # set description "inside link"
Firepower /ssa/logical-device/external-port-link* # exit
Firepower /ssa/logical-device* # create external-port-link management Ethernet1/7 asa
Firepower /ssa/logical-device/external-port-link* # set description "management link"
Firepower /ssa/logical-device/external-port-link* # exit
Firepower /ssa/logical-device* # create external-port-link outside Ethernet1/2 asa
Firepower /ssa/logical-device/external-port-link* # set description "external link"
Firepower /ssa/logical-device/external-port-link* # exit
```

Step 5 Configure the management bootstrap information.

a) Create the bootstrap object.

create mgmt-bootstrap asa

Example:

```
Firepower /ssa/logical-device* # create mgmt-bootstrap asa
Firepower /ssa/logical-device/mgmt-bootstrap* #
```

b) Specify the admin password.

create bootstrap-key-secret PASSWORD

set value

Enter a value: *password*

Confirm the value: *password*

exit

Example:

The pre-configured ASA admin user is useful for password recovery; if you have FXOS access, you can reset the admin user password if you forget it.

Example:

```
Firepower /ssa/logical-device/mgmt-bootstrap* # create bootstrap-key-secret PASSWORD
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key-secret* # set value
Enter a value: floppylampshade
Confirm the value: floppylampshade
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key-secret* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* #
```

c) Configure the IPv4 management interface settings.

create ipv4 *slot_id* **default**

set ip *ip_address* **mask** *network_mask*

set gateway *gateway_address*

exit

Example:

```
Firepower /ssa/logical-device/mgmt-bootstrap* # create ipv4 1 default
Firepower /ssa/logical-device/mgmt-bootstrap/ipv4* # set ip 10.10.10.34 mask 255.255.255.0
Firepower /ssa/logical-device/mgmt-bootstrap/ipv4* # set gateway 10.10.10.1
Firepower /ssa/logical-device/mgmt-bootstrap/ipv4* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* #
```

d) Configure the IPv6 management interface settings.

create ipv6 *slot_id* **default**

set ip *ip_address* **prefix-length** *prefix*

set gateway *gateway_address*

exit

Example:

```
Firepower /ssa/logical-device/mgmt-bootstrap* # create ipv6 1 default
Firepower /ssa/logical-device/mgmt-bootstrap/ipv6* # set ip 2001:0DB8:BA98::3210
prefix-length 64
Firepower /ssa/logical-device/mgmt-bootstrap/ipv6* # set gateway 2001:0DB8:BA98::3211
Firepower /ssa/logical-device/mgmt-bootstrap/ipv6* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* #
```

e) Exit the management bootstrap mode.

exit

Example:

```
Firepower /ssa/logical-device/mgmt-bootstrap* # exit
Firepower /ssa/logical-device* #
```

Step 6 Save the configuration.

commit-buffer

The chassis deploys the logical device by downloading the specified software version and pushing the bootstrap configuration and management interface settings to the application instance. Check the status of the deployment using the **show app-instance** command. The application instance is running and ready to use when the **Admin State** is **Enabled** and the **Oper State** is **Online**.

Example:

```
Firepower /ssa/logical-device* # commit-buffer
Firepower /ssa/logical-device # exit
Firepower /ssa # show app-instance
App Name   Identifier Slot ID    Admin State Oper State      Running Version Startup Version
 Deploy Type Profile Name Cluster State   Cluster Role
---------- ---------- ---------- ----------- ---------------- --------------- ---------------
 ----------- ------------ --------------- ------------
asa        asa1       2          Disabled    Not Installed                    9.12.1
    Native                   Not Applicable  None
ftd        ftd1       1          Enabled     Online           6.4.0.49        6.4.0.49
    Container   Default-Small Not Applicable  None
```

Step 7 See the ASA configuration guide to start configuring your security policy.

Example

```
Firepower# scope ssa
Firepower /ssa # scope slot 1
Firepower /ssa/slot # enter app-instance asa
Firepower /ssa/slot/app-instance* # set startup-version 9.10.1
Firepower /ssa/slot/app-instance* # exit
Firepower /ssa/slot* # exit
Firepower /ssa* # create logical-device MyDevice1 asa 1 standalone
Firepower /ssa/logical-device* # create external-port-link inside Ethernet1/1 asa
Firepower /ssa/logical-device/external-port-link* # set description "inside link"
Firepower /ssa/logical-device/external-port-link* # exit
Firepower /ssa/logical-device* # create external-port-link management Ethernet1/7 asa
Firepower /ssa/logical-device/external-port-link* # set description "management link"
Firepower /ssa/logical-device/external-port-link* # exit
Firepower /ssa/logical-device* # create external-port-link outside Ethernet1/2 asa
Firepower /ssa/logical-device/external-port-link* # set description "external link"
Firepower /ssa/logical-device/external-port-link* # exit
Firepower /ssa/logical-device* # create mgmt-bootstrap asa
Firepower /ssa/logical-device/mgmt-bootstrap* # create bootstrap-key-secret PASSWORD
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key-secret* # set value
Enter a value: secretglassine
Confirm the value: secretglassine
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key-secret* # exit
```

```
Firepower /ssa/logical-device/mgmt-bootstrap* # create ipv4 1 default
Firepower /ssa/logical-device/mgmt-bootstrap/ipv4* # set gateway 10.0.0.1
Firepower /ssa/logical-device/mgmt-bootstrap/ipv4* # set ip 10.0.0.31 mask 255.255.255.0
Firepower /ssa/logical-device/mgmt-bootstrap/ipv4* # exit
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # commit-buffer
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key #
```

Add a Standalone Firepower Threat Defense

Standalone logical devices work either alone or in a High Availability pair. On the Firepower 9300 with multiple security modules, you can deploy either a cluster or standalone devices. The cluster must use all modules, so you cannot mix and match a 2-module cluster plus a single standalone device, for example.

Before you begin

- Download the application image you want to use for the logical device from Cisco.com, and then download that image to the Firepower 4100/9300 chassis.

  Note For the Firepower 9300, you must install the same application instance type (ASA or FTD) on all modules in the chassis; different types are not supported at this time. Note that modules can run different versions of an application instance type.

- Configure a management interface to use with the logical device. The management interface is required. Note that this management interface is not the same as the chassis management port that is used only for chassis management (in FXOS, you might see it displayed as MGMT, management0, or other similar names).
- You must also configure at least one Data type interface. Optionally, you can also create a firepower-eventing interface to carry all event traffic (such as web events). See Interface Types, on page 105 for more information.
- Gather the following information:
  - Interface IDs for this device
  - Management interface IP address and network mask
  - Gateway IP address
  - FMC IP address and/or NAT ID of your choosing
  - DNS server IP address
  - FTD hostname and domain name

Procedure

Step 1 Enter security services mode.

scope ssa

Example:

```
Firepower# scope ssa
Firepower /ssa #
```

Step 2 Accept the end-user license agreement for the Firepower Threat Defense version you want to use. You only need to perform this step if you have not already accepted the EULA for this version.

a) View available images. Note the Version number that you want to use.

show app

Example:

```
Firepower /ssa # show app
    Name       Version         Author     Supported Deploy Types CSP Type    Is Default
 App
    ---------- --------------- ---------- ---------------------- -----------
--------------
    asa        9.9.1           cisco      Native                 Application No
    asa        9.10.1          cisco      Native                 Application Yes
    ftd        6.2.3           cisco      Native                 Application Yes
```

b) Set the scope to the image version.

scope app ftd *application_version*

Example:

```
Firepower /ssa # scope app ftd 6.2.3
Firepower /ssa/app #
```

c) Accept the license agreement.

accept-license-agreement

Example:

```
Firepower /ssa/app # accept-license-agreement

End User License Agreement: End User License Agreement

Effective: May 22, 2017

This is an agreement between You and Cisco Systems, Inc. or its affiliates
("Cisco") and governs your Use of Cisco Software. "You" and "Your" means the
individual or legal entity licensing the Software under this EULA. "Use" or
"Using" means to download, install, activate, access or otherwise use the
Software. "Software" means the Cisco computer programs and any Upgrades made
available to You by an Approved Source and licensed to You by Cisco.
"Documentation" is the Cisco user or technical manuals, training materials,
specifications or other documentation applicable to the Software and made
available to You by an Approved Source. "Approved Source" means (i) Cisco or
(ii) the Cisco authorized reseller, distributor or systems integrator from whom
you acquired the Software. "Entitlement" means the license detail; including
license metric, duration, and quantity provided in a product ID (PID) published
on Cisco's price list, claim certificate or right to use notification.
"Upgrades" means all updates, upgrades, bug fixes, error corrections,
enhancements and other modifications to the Software and backup copies thereof.
```

```
[...]

Please "commit-buffer" if you accept the license agreement, otherwise "discard-buffer".

Firepower /ssa/app* #
```

d) Save the configuration.

commit-buffer

Example:

```
Firepower /ssa/app* # commit-buffer
Firepower /ssa/app #
```

e) Exit to security services mode.

exit

Example:

```
Firepower /ssa/app # exit
Firepower /ssa #
```

Step 3 Set the application instance image version.

a) Set the scope to the security module/engine slot.

scope slot *slot_id*

The *slot_id* is always 1 for the Firepower 4100, and 1, 2, or 3 for the Firepower 9300.

Example:

```
Firepower /ssa # scope slot 1
Firepower /ssa/slot #
```

b) Create the application instance.

enter app-instance ftd

Example:

```
Firepower /ssa/slot # enter app-instance ftd
Firepower /ssa/slot/app-instance* #
```

c) Set the Firepower Threat Defense image version.

set startup-version *version*

Enter the version number that you noted earlier in this procedure when you accepted the EULA.

Example:

```
Firepower /ssa/slot/app-instance* # set startup-version 6.3.0
```

d) Exit to slot mode.

exit

Example:

```
Firepower /ssa/slot/app-instance* # exit
Firepower /ssa/slot* #
```

e) Exit to ssa mode.

exit

Example:

```
Firepower /ssa/slot* # exit
Firepower /ssa* #
```

Example:

```
Firepower /ssa # scope slot 1
Firepower /ssa/slot # enter app-instance ftd MyDevice1
Firepower /ssa/slot/app-instance* # set startup-version 6.3.0
Firepower /ssa/slot/app-instance* # exit
Firepower /ssa/slot* # exit
Firepower /ssa* #
```

Step 4 Create the logical device.

enter logical-device *device_name* **ftd** *slot_id* **standalone**

Example:

```
Firepower /ssa # enter logical-device FTD1 ftd 1 standalone
Firepower /ssa/logical-device* #
```

Step 5 Assign the management and data interfaces to the logical device. Repeat for each interface.

create external-port-link *name interface_id* **ftd**

set description *description*

exit

- *name*—The name is used by the Firepower 4100/9300 chassis supervisor; it is not the interface name used in the Firepower Threat Defense configuration.
- *description*—Use quotes (") around phrases with spaces.

The management interface is not the same as the chassis management port. You will later enable and configure the data interfaces in FMC, including setting the IP addresses.

Example:

```
Firepower /ssa/logical-device* # create external-port-link inside Ethernet1/1 ftd
Firepower /ssa/logical-device/external-port-link* # set description "inside link"
Firepower /ssa/logical-device/external-port-link* # exit
Firepower /ssa/logical-device* # create external-port-link management Ethernet1/7 ftd
Firepower /ssa/logical-device/external-port-link* # set description "management link"
Firepower /ssa/logical-device/external-port-link* # exit
```

```
Firepower /ssa/logical-device* # create external-port-link outside Ethernet1/2 ftd
Firepower /ssa/logical-device/external-port-link* # set description "external link"
Firepower /ssa/logical-device/external-port-link* # exit
```

Step 6 Configure the management bootstrap parameters.

These settings are meant for initial deployment only, or for disaster recovery. For normal operation, you can later change most values in the application CLI configuration.

a) Create the bootstrap object.

create mgmt-bootstrap ftd

Example:

```
Firepower /ssa/logical-device* # create mgmt-bootstrap ftd
Firepower /ssa/logical-device/mgmt-bootstrap* #
```

b) Specify the IP address or hostname of the managing Firepower Management Center:

Set one of the following:

- **enter bootstrap-key FIREPOWER_MANAGER_IP**

  set value *IP_address*

  exit

- **enter bootstrap-key FQDN**

  set value *fmc_hostname*

  exit

Example:

```
Firepower /ssa/logical-device/mgmt-bootstrap* # create bootstrap-key FIREPOWER_MANAGER_IP
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # set value 10.10.10.7
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* #
```

c) Specify the firewall mode, routed or transparent.

create bootstrap-key FIREWALL_MODE

set value {**routed** | **transparent**}

exit

In routed mode, the device is considered to be a router hop in the network. Each interface that you want to route between is on a different subnet. A transparent firewall, on the other hand, is a Layer 2 firewall that acts like a “bump in the wire,” or a “stealth firewall,” and is not seen as a router hop to connected devices.

The firewall mode is only set at initial deployment. If you re-apply the bootstrap settings, this setting is not used.

Example:

```
Firepower /ssa/logical-device/mgmt-bootstrap* # create bootstrap-key FIREWALL_MODE
```

```
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # set value routed
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* #
```

d) Specify the key to be shared between the device and the Firepower Management Center. You can choose any passphrase for this key between 1 and 37 characters; you will enter the same key on the FMC when you add the FTD.

create bootstrap-key-secret REGISTRATION_KEY

set value

Enter a value: *registration_key*

Confirm the value: *registration_key*

exit

Example:

```
Firepower /ssa/logical-device/mgmt-bootstrap* # create bootstrap-key-secret
REGISTRATION_KEY
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key-secret* # set value
Enter a value: gratuitousapples
Confirm the value: gratuitousapples
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key-secret* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* #
```

e) Specify the admin password. This password is used for the admin user for CLI access.

create bootstrap-key-secret PASSWORD

set value

Enter a value: *password*

Confirm the value: *password*

exit

Example:

```
Firepower /ssa/logical-device/mgmt-bootstrap* # create bootstrap-key-secret PASSWORD
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key-secret* # set value
Enter a value: floppylampshade
Confirm the value: floppylampshade
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key-secret* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* #
```

f) Specify the fully qualified hostname.

create bootstrap-key FQDN

set value *fqdn*

exit

Example:

```
Firepower /ssa/logical-device/mgmt-bootstrap* # create bootstrap-key FQDN
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # set value ftd1.cisco.com
```

```
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* #
```

g) Specify a comma-separated list of DNS servers.

create bootstrap-key DNS_SERVERS

set value *dns_servers*

exit

The FTD uses DNS if you specify a hostname for the FMC, for example.

Example:

```
Firepower /ssa/logical-device/mgmt-bootstrap* # create bootstrap-key DNS_SERVERS
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # set value 10.9.8.7,10.9.6.5
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* #
```

h) Specify a comma-separated list of search domains.

create bootstrap-key SEARCH_DOMAINS

set value *search_domains*

exit

Example:

```
Firepower /ssa/logical-device/mgmt-bootstrap* # create bootstrap-key SEARCH_DOMAINS
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # set value
cisco.com,example.com
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* #
```

i) Configure the IPv4 management interface settings.

create ipv4 *slot_id* **firepower**

set ip *ip_address* **mask** *network_mask*

set gateway *gateway_address*

exit

Example:

```
Firepower /ssa/logical-device/mgmt-bootstrap* # create ipv4 1 firepower
Firepower /ssa/logical-device/mgmt-bootstrap/ipv4* # set ip 10.10.10.34 mask 255.255.255.0
Firepower /ssa/logical-device/mgmt-bootstrap/ipv4* # set gateway 10.10.10.1
Firepower /ssa/logical-device/mgmt-bootstrap/ipv4* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* #
```

j) Configure the IPv6 management interface settings.

create ipv6 *slot_id* **firepower**

set ip *ip_address* **prefix-length** *prefix*

set gateway *gateway_address*

exit

Example:

```
Firepower /ssa/logical-device/mgmt-bootstrap* # create ipv6 1 firepower
Firepower /ssa/logical-device/mgmt-bootstrap/ipv6* # set ip 2001:0DB8:BA98::3210
prefix-length 64
Firepower /ssa/logical-device/mgmt-bootstrap/ipv6* # set gateway 2001:0DB8:BA98::3211
Firepower /ssa/logical-device/mgmt-bootstrap/ipv6* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* #
```

k) Exit the management bootstrap mode.

exit

Example:

```
Firepower /ssa/logical-device/mgmt-bootstrap* # exit
Firepower /ssa/logical-device* #
```

Step 7 Save the configuration.

commit-buffer

The chassis deploys the logical device by downloading the specified software version and pushing the bootstrap configuration and management interface settings to the application instance. Check the status of the deployment using the **show app-instance** command. The application instance is running and ready to use when the **Admin State** is **Enabled** and the **Oper State** is **Online**.

Example:

```
Firepower /ssa/logical-device* # commit-buffer
Firepower /ssa/logical-device # exit
Firepower /ssa # show app-instance
App Name   Identifier Slot ID    Admin State Oper State      Running Version Startup Version
 Deploy Type Profile Name Cluster State   Cluster Role
---------- ---------- ---------- ----------- --------------- --------------- ---------------
 ----------- ------------ --------------- ------------
asa        asa1       2          Disabled    Not Installed                   9.12.1
    Native                   Not Applicable  None
ftd        ftd1       1          Enabled     Online          6.4.0.49        6.4.0.49
    Container   Default-Small Not Applicable  None
```

Step 8 See the FMC configuration guide to add the FTD as a managed device and start configuring your security policy.

Example

```
Firepower# scope ssa
Firepower /ssa* # scope app ftd 6.3.0
Firepower /ssa/app* # accept-license-agreement
Firepower /ssa/app* # commit-buffer
Firepower /ssa/app # exit
Firepower /ssa # scope slot 1
Firepower /ssa/slot # enter app-instance ftd
```

```
Firepower /ssa/slot/app-instance* # set startup-version 6.3.0
Firepower /ssa/slot/app-instance* # exit
Firepower /ssa/slot* # exit
Firepower /ssa* # create logical-device MyDevice1 ftd 1 standalone
Firepower /ssa/logical-device* # create external-port-link inside Ethernet1/1 ftd
Firepower /ssa/logical-device/external-port-link* # set description "inside link"
Firepower /ssa/logical-device/external-port-link* # exit
Firepower /ssa/logical-device* # create external-port-link management Ethernet1/7 ftd
Firepower /ssa/logical-device/external-port-link* # set description "management link"
Firepower /ssa/logical-device/external-port-link* # exit
Firepower /ssa/logical-device* # create external-port-link outside Ethernet1/2 ftd
Firepower /ssa/logical-device/external-port-link* # set description "external link"
Firepower /ssa/logical-device/external-port-link* # exit
Firepower /ssa/logical-device* # create mgmt-bootstrap ftd
Firepower /ssa/logical-device/mgmt-bootstrap* # create bootstrap-key FIREPOWER_MANAGER_IP
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # set value 10.0.0.100
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* # create bootstrap-key FIREWALL_MODE
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # set value routed
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* # create bootstrap-key-secret REGISTRATION_KEY
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key-secret* # set value
Enter a value: juniorwindowpane
Confirm the value: juniorwindowpane
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key-secret* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* # create bootstrap-key-secret PASSWORD
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key-secret* # set value
Enter a value: secretglassine
Confirm the value: secretglassine
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key-secret* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* # create ipv4 1 firepower
Firepower /ssa/logical-device/mgmt-bootstrap/ipv4* # set gateway 10.0.0.1
Firepower /ssa/logical-device/mgmt-bootstrap/ipv4* # set ip 10.0.0.31 mask 255.255.255.0
Firepower /ssa/logical-device/mgmt-bootstrap/ipv4* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* # create bootstrap-key FQDN
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # set value ftd.cisco.com
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* # create bootstrap-key DNS_SERVERS
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # set value 192.168.1.1
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* # create bootstrap-key SEARCH_DOMAINS
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # set value search.com
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # commit-buffer
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key #
```

Add a High Availability Pair

or High Availability (also known as failover) is configured within the application, not in FXOS. However, to prepare your chassis for high availability, see the following steps.

Before you begin

See .

Procedure

Step 1 Allocate the same interfaces to each logical device.

Step 2 Allocate 1 or 2 data interfaces for the failover and state link(s).

These interfaces exchange high availability traffic between the 2 chassis. We recommend that you use a 10 GB data interface for a combined failover and state link. If you have available interfaces, you can use separate failover and state links; the state link requires the most bandwidth. You cannot use the management-type interface for the failover or state link. We recommend that you use a switch between the chassis, with no other device on the same network segment as the failover interfaces.

Step 3 Enable High Availability on the logical devices.

Step 4 If you need to make interface changes after you enable High Availability, perform the changes on the standby unit first, and then perform the changes on the active unit.

Add a Cluster

Clustering lets you group multiple devices together as a single logical device. A cluster provides all the convenience of a single device (management, integration into a network) while achieving the increased throughput and redundancy of multiple devices. The Firepower 9300, which includes multiple modules, supports intra-chassis clustering where you group all modules within a single chassis into a cluster. You can also use inter-chassis clustering, where multiple chassis are grouped together; inter-chasis clustering is the only option for single module devices like the Firepower 4100 series.

Note The FTD does not support a cluster across multiple chassis (inter-chassis); only intra-chassis clustering is supported.

About Clustering on the Firepower 4100/9300 Chassis

The cluster consists of multiple devices acting as a single logical unit. When you deploy a cluster on the Firepower 4100/9300 chassis, it does the following:

- Creates a *cluster-control link* (by default, port-channel 48) for unit-to-unit communication.

  For intra-chassis clustering (Firepower 9300 only), this link utilizes the Firepower 9300 backplane for cluster communications.

  For inter-chassis clustering, you need to manually assign physical interface(s) to this EtherChannel for communications between chassis.

- Creates the cluster bootstrap configuration within the application.

  When you deploy the cluster, the chassis supervisor pushes a minimal bootstrap configuration to each unit that includes the cluster name, cluster control link interface, and other cluster settings. Some parts of the bootstrap configuration may be user-configurable within the application if you want to customize your clustering environment.

- Assigns data interfaces to the cluster as *Spanned* interfaces.

  For intra-chassis clustering, spanned interfaces are not limited to EtherChannels, like it is for inter-chassis clustering.The Firepower 9300 supervisor uses EtherChannel technology internally to load-balance traffic to multiple modules on a shared interface, so any data interface type works for Spanned mode. For inter-chassis clustering, you must use Spanned EtherChannels for all data interfaces.

  Note Individual interfaces are not supported, with the exception of a management interface.

- Assigns a management interface to all units in the cluster.

The following sections provide more detail about clustering concepts and implementation.

Primary and Secondary Unit Roles

One member of the cluster is the primary unit. The primary unit is determined automatically. All other members are secondary units.

You must perform all configuration on the primary unit only; the configuration is then replicated to the secondary units.

Cluster Control Link

The cluster control link is automatically created using the Port-channel 48 interface.

For intra-chassis clustering, this interface has no member interfaces. This Cluster type EtherChannel utilizes the Firepower 9300 backplane for cluster communications for intra-chassis clustering. For inter-chassis clustering, you must add one or more interfaces to the EtherChannel.

For a 2-member inter-chassis cluster, do not directly connect the cluster control link from one chassis to the other chassis. If you directly connect the interfaces, then when one unit fails, the cluster control link fails, and thus the remaining healthy unit fails. If you connect the cluster control link through a switch, then the cluster control link remains up for the healthy unit.

Cluster control link traffic includes both control and data traffic.

Size the Cluster Control Link for Inter-Chassis Clustering

If possible, you should size the cluster control link to match the expected throughput of each chassis so the cluster-control link can handle the worst-case scenarios.

Cluster control link traffic is comprised mainly of state update and forwarded packets. The amount of traffic at any given time on the cluster control link varies. The amount of forwarded traffic depends on the load-balancing efficacy or whether there is a lot of traffic for centralized features. For example:

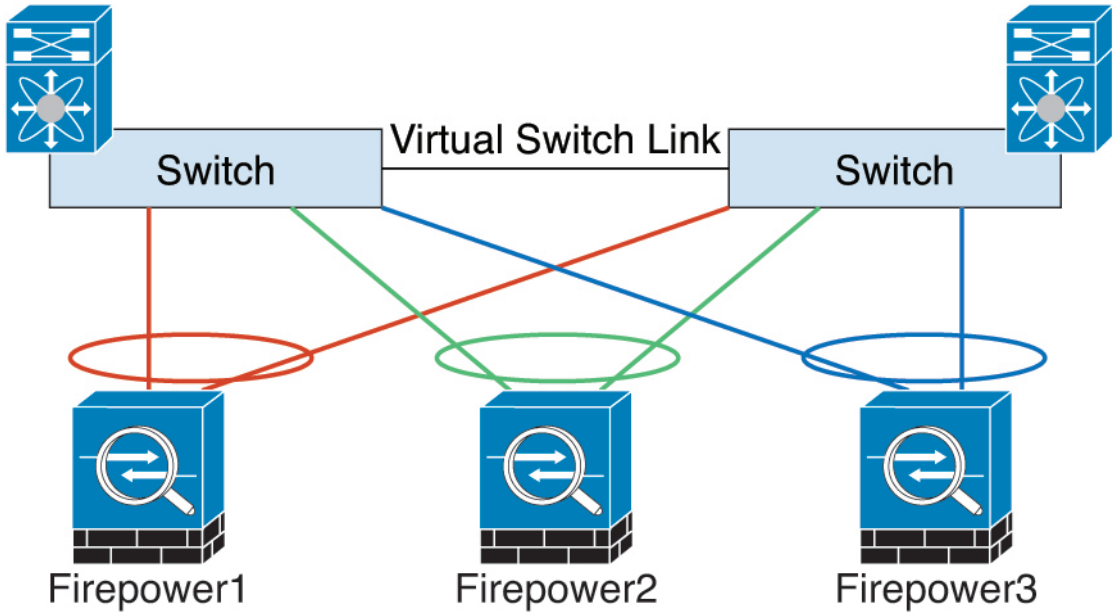
- NAT results in poor load balancing of connections, and the need to rebalance all returning traffic to the correct units.
- When membership changes, the cluster needs to rebalance a large number of connections, thus temporarily using a large amount of cluster control link bandwidth.

A higher-bandwidth cluster control link helps the cluster to converge faster when there are membership changes and prevents throughput bottlenecks.

Note If your cluster has large amounts of asymmetric (rebalanced) traffic, then you should increase the cluster control link size.

Cluster Control Link Redundancy for Inter-Chassis Clustering

The following diagram shows how to use an EtherChannel as a cluster control link in a Virtual Switching System (VSS) or Virtual Port Channel (vPC) environment. All links in the EtherChannel are active. When the switch is part of a VSS or vPC, then you can connect Firepower 4100/9300 chassis interfaces within the same EtherChannel to separate switches in the VSS or vPC. The switch interfaces are members of the same EtherChannel port-channel interface, because the separate switches act like a single switch. Note that this EtherChannel is device-local, not a Spanned EtherChannel.



Cluster Control Link Reliability for Inter-Chassis Clustering

To ensure cluster control link functionality, be sure the round-trip time (RTT) between units is less than 20 ms. This maximum latency enhances compatibility with cluster members installed at different geographical sites. To check your latency, perform a ping on the cluster control link between units.

The cluster control link must be reliable, with no out-of-order or dropped packets; for example, for inter-site deployment, you should use a dedicated link.

Cluster Control Link Network

The Firepower 4100/9300 chassis auto-generates the cluster control link interface IP address for each unit based on the chassis ID and slot ID: 127.2.*chassis_id*.*slot_id*. The cluster control link network cannot include any routers between units; only Layer 2 switching is allowed. For inter-site traffic, Cisco recommends using Overlay Transport Virtualization (OTV).

Management Network

We recommend connecting all units to a single management network. This network is separate from the cluster control link.

Management Interface

You must assign a Management type interface to the cluster. This interface is a special *individual* interface as opposed to a Spanned interface. The management interface lets you connect directly to each unit.

For the ASA, the Main cluster IP address is a fixed address for the cluster that always belongs to the current primary unit. You must configure a range of addresses so that each unit, including the current primary unit, can use a Local address from the range. The Main cluster IP address provides consistent management access to an address; when a primary unit changes, the Main cluster IP address moves to the new primary unit, so management of the cluster continues seamlessly. The Local IP address is used for routing, and is also useful for troubleshooting. For example, you can manage the cluster by connecting to the Main cluster IP address, which is always attached to the current primary unit. To manage an individual member, you can connect to the Local IP address. For outbound management traffic such as TFTP or syslog, each unit, including the primary unit, uses the Local IP address to connect to the server.

For the Firepower Threat Defense, assign a management IP address to each unit on the same network. Use these IP addresses when you add each unit to the FMC.

Spanned EtherChannels

You can group one or more interfaces per chassis into an EtherChannel that spans all chassis in the cluster. The EtherChannel aggregates the traffic across all the available active interfaces in the channel. A Spanned EtherChannel can be configured in both routed and transparent firewall modes. In routed mode, the EtherChannel is configured as a routed interface with a single IP address. In transparent mode, the IP address is assigned to the BVI, not to the bridge group member interface. The EtherChannel inherently provides load balancing as part of basic operation.

Inter-Site Clustering

For inter-site installations, you can take advantage of clustering as long as you follow the recommended guidelines.

You can configure each cluster chassis to belong to a separate site ID.

Site IDs work with site-specific MAC addresses and IP addresses. Packets sourced from the cluster use a site-specific MAC address and IP address, while packets received by the cluster use a global MAC address and IP address. This feature prevents the switches from learning the same global MAC address from both sites on two different ports, which causes MAC flapping; instead, they only learn the site MAC address. Site-specific MAC addresses and IP address are supported for routed mode using Spanned EtherChannels only.

Site IDs are also used to enable flow mobility using LISP inspection.

See the following sections for more information about inter-site clustering:

- Sizing the Data Center Interconnect—Requirements and Prerequisites for Clustering, on page 116
- Inter-Site Guidelines—Clustering Guidelines and Limitations, on page 119
- Inter-Site Examples—Examples for Inter-Site Clustering, on page 177

Add an ASA Cluster

You can add a single Firepower 9300 chassis as an intra-chassis cluster, or add multiple chassis for inter-chassis clustering. For inter-chassis clustering, you must configure each chassis separately. Add the cluster on one chassis; you can then enter most of the same settings on the next chassis.

Create an ASA Cluster

Set the scope to the image version.

You can easily deploy the cluster from the Firepower 4100/9300 chassis supervisor. All initial configuration is automatically generated for each unit.

For inter-chassis clustering, you must configure each chassis separately. Deploy the cluster on one chassis; you can then copy the bootstrap configuration from the first chassis to the next chassis for ease of deployment.

In a Firepower 9300 chassis, you must enable clustering for all 3 module slots, even if you do not have a module installed. If you do not configure all 3 modules, the cluster will not come up.

For multiple context mode, you must first deploy the logical device, and then enable multiple context mode in the ASA application.

Before you begin

- Download the application image you want to use for the logical device from Cisco.com, and then upload that image to the Firepower 4100/9300 chassis.
- Gather the following information:
  - Management interface ID, IP address, and network mask
  - Gateway IP address

Procedure

Step 1 Configure interfaces.

a) Add at least one Data type interface or EtherChannel (also known as a port-channel) before you deploy the cluster. See Add an EtherChannel (Port Channel), on page 109 or Configure a Physical Interface, on page 107.

For inter-chassis clustering, all data interfaces must be Spanned EtherChannels with at least one member interface. Add the same EtherChannels on each chassis. Combine the member interfaces from all cluster units into a single EtherChannel on the switch. See Clustering Guidelines and Limitations, on page 119 for more information about EtherChannels for inter-chassis clustering.

b) Add a Management type interface or EtherChannel. See Add an EtherChannel (Port Channel), on page 109 or Configure a Physical Interface, on page 107.

The management interface is required. Note that this management interface is not the same as the chassis management interface that is used only for chassis management (in FXOS, you might see the chassis management interface displayed as MGMT, management0, or other similar names).

For inter-chassis clustering, add the same Management interface on each chassis.

c) For inter-chassis clustering, add a member interface to the cluster control link EtherChannel (by default, port-channel 48). See Add an EtherChannel (Port Channel), on page 109.

Do not add a member interface for intra-chassis clustering. If you add a member, the chassis assumes this cluster will be inter-chassis, and will only allow you to use Spanned EtherChannels, for example.

Add the same member interfaces on each chassis. The cluster control link is a device-local EtherChannel on each chassis. Use separate EtherChannels on the switch per device. See Clustering Guidelines and Limitations, on page 119 for more information about EtherChannels for inter-chassis clustering.

Step 2 Enter security services mode.

scope ssa

Example:

```
Firepower# scope ssa
Firepower /ssa #
```

Step 3 Set the application instance image version.

a) View available images. Note the Version number that you want to use.

show app

Example:

```
Firepower /ssa # show app
    Name       Version         Author     Supported Deploy Types CSP Type    Is Default
 App
    ---------- --------------- ---------- ---------------------- -----------
--------------
    asa        9.9.1           cisco      Native                 Application No
    asa        9.10.1          cisco      Native                 Application Yes
    ftd        6.2.3           cisco      Native                 Application Yes
```

b) Set the scope to the image version.

scope app asa *application_version*

Example:

```
Firepower /ssa # scope app asa 9.10.1
Firepower /ssa/app #
```

c) Set this version as the default.

set-default

Example:

```
Firepower /ssa/app # set-default
Firepower /ssa/app* #
```

d) Exit to ssa mode.

exit

Example:

```
Firepower /ssa/app* # exit
Firepower /ssa* #
```

Example:

```
Firepower /ssa # scope app asa 9.12.1
Firepower /ssa/app # set-default
Firepower /ssa/app* # exit
Firepower /ssa* #
```

Step 4 Create the cluster.

enter logical-device *device_name* **asa** *slots* **clustered**

- *device_name*—Used by the Firepower 4100/9300 chassis supervisor to configure clustering settings and assign interfaces; it is not the cluster name used in the security module configuration. You must specify all three security modules, even if you have not yet installed the hardware.
- *slots*—Assigns the chassis modules to the cluster. For the Firepower 4100, specify **1**. For the Firepower 9300, specify **1,2,3**. You must enable clustering for all 3 module slots in a Firepower 9300 chassis, even if you do not have a module installed. If you do not configure all 3 modules, the cluster will not come up.

Example:

```
Firepower /ssa # enter logical-device ASA1 asa 1,2,3 clustered
Firepower /ssa/logical-device* #
```

Step 5 Configure the cluster bootstrap parameters.

These settings are meant for initial deployment only, or for disaster recovery. For normal operation, you can later change most values in the application CLI configuration.

a) Create the cluster bootstrap object.

enter cluster-bootstrap

Example:

```
Firepower /ssa/logical-device* # enter cluster-bootstrap
Firepower /ssa/logical-device/cluster-bootstrap* #
```

b) Configure an authentication key for control traffic on the cluster control link.

set key

Example:

```
Firepower /ssa/logical-device/cluster-bootstrap* # set key
Key: diamonddogs
```

You are prompted to enter the shared secret.

The shared secret is an ASCII string from 1 to 63 characters. The shared secret is used to generate the key. This option does not affect datapath traffic, including connection state update and forwarded packets, which are always sent in the clear.

c) Set the cluster interface mode.

set mode spanned-etherchannel

Spanned EtherChannel mode is the only supported mode.

Example:

```
Firepower /ssa/logical-device/cluster-bootstrap* # set mode spanned-etherchannel
Firepower /ssa/logical-device/cluster-bootstrap* #
```

d) Set the cluster group name in the security module configuration.

set service-type *cluster_name*

The name must be an ASCII string from 1 to 38 characters.

Example:

```
Firepower /ssa/logical-device/cluster-bootstrap* # set service-type cluster1
Firepower /ssa/logical-device/cluster-bootstrap* #
```

e) Configure the management IP address information.

This information is used to configure a management interface in the security module configuration.

1. Configure a pool of Local IP addresses, one of which will be assigned to each cluster unit for the interface.

   set ipv4 pool *start_ip end_ip*

   set ipv6 pool *start_ip end_ip*

   Include at least as many addresses as there are units in the cluster. Note that for the Firepower 9300, you must include 3 addresses per chassis, even if you do not have all module slots filled. If you plan to expand the cluster, include additional addresses. The Virtual IP address (known as the Main cluster IP address) that belongs to the current control unit is *not* a part of this pool; be sure to reserve an IP address on the same network for the Main cluster IP address. You can use IPv4 and/or IPv6 addresses.

2. Configure the Main cluster IP address for the management interface.

   set virtual ipv4 *ip_address* **mask** *mask*

   set virtual ipv6 *ip_address* **prefix-length** *prefix*

   This IP address must be on the same network as the cluster pool addresses, but not be part of the pool.

3. Enter the network gateway address.

   set ipv4 gateway *ip_address*

   set ipv6 gateway *ip_address*

Example:

```
Firepower /ssa/logical-device/cluster-bootstrap* # set ipv4 gateway 10.1.1.254
```

```
Firepower /ssa/logical-device/cluster-bootstrap* # set ipv4 pool 10.1.1.11 10.1.1.27
Firepower /ssa/logical-device/cluster-bootstrap* # set ipv6 gateway 2001:DB8::AA
Firepower /ssa/logical-device/cluster-bootstrap* # set ipv6 pool 2001:DB8::11 2001:DB8::27
Firepower /ssa/logical-device/cluster-bootstrap* # set virtual ipv4 10.1.1.1 mask
255.255.255.0
Firepower /ssa/logical-device/cluster-bootstrap* # set virtual ipv6 2001:DB8::1
prefix-length 64
```

f) Exit the cluster bootstrap mode.

exit

Example:

```
Firepower /ssa/logical-device* # enter cluster-bootstrap
Firepower /ssa/logical-device/cluster-bootstrap* # set chassis-id 1
Firepower /ssa/logical-device/cluster-bootstrap* # set key
  Key: f@arscape
Firepower /ssa/logical-device/cluster-bootstrap* # set mode spanned-etherchannel
Firepower /ssa/logical-device/cluster-bootstrap* # set service-type cluster1
Firepower /ssa/logical-device/cluster-bootstrap* # exit
Firepower /ssa/logical-device/* #
```

Step 6 Configure the management bootstrap parameters.

These settings are meant for initial deployment only, or for disaster recovery. For normal operation, you can later change most values in the application CLI configuration.

a) Create the management bootstrap object.

enter mgmt-bootstrap asa

Example:

```
Firepower /ssa/logical-device* # enter mgmt-bootstrap asa
Firepower /ssa/logical-device/mgmt-bootstrap* #
```

b) Specify the admin password.

create bootstrap-key-secret PASSWORD

set value

Enter a value: *password*

Confirm the value: *password*

exit

Example:

The pre-configured ASA admin user is useful for password recovery; if you have FXOS access, you can reset the admin user password if you forget it.

Example:

```
Firepower /ssa/logical-device/mgmt-bootstrap* # create bootstrap-key-secret PASSWORD
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key-secret* # set value
Enter a value: floppylampshade
Confirm the value: floppylampshade
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key-secret* # exit
```

```
Firepower /ssa/logical-device/mgmt-bootstrap* #
```

c) Exit the management bootstrap mode.

exit

Example:

```
Firepower /ssa/logical-device/mgmt-bootstrap* # exit
Firepower /ssa/logical-device* #
```

Step 7 Save the configuration.

commit-buffer

The chassis deploys the logical device by downloading the specified software version and pushing the bootstrap configuration and management interface settings to the application instance. Check the status of the deployment using the **show app-instance** command. The application instance is running and ready to use when the **Admin State** is **Enabled** and the **Oper State** is **Online**.

Example:

```
Firepower /ssa/logical-device* # commit-buffer
Firepower /ssa/logical-device # exit
Firepower /ssa # show app-instance
App Name   Identifier Slot ID    Admin State Oper State      Running Version Startup Version
 Deploy Type Profile Name Cluster State   Cluster Role
---------- ---------- ---------- ----------- ---------------- --------------- ---------------
 ----------- ------------ --------------- ------------
ftd        cluster1   1          Enabled     Online           6.4.0.49        6.4.0.49
    Native                    In Cluster      Slave
ftd        cluster1   2          Enabled     Online           6.4.0.49        6.4.0.49
    Native                    In Cluster      Master
ftd        cluster1   3          Disabled    Not Available                    6.4.0.49
    Native                    Not Applicable  None
```

Step 8 To add another chassis to the cluster, repeat this procedure except you must configure a unique **chassis-id**; otherwise, use the same configuration for both chassis.

Make sure the interface configuration is the same on the new chassis. You can export and import FXOS chassis configuration to make this process easier.

Step 9 Connect to the control unit ASA to customize your clustering configuration.

Example

For chassis 1:

```
scope eth-uplink
  scope fabric a
    enter port-channel 1
      set port-type data
      enable
      enter member-port Ethernet1/1
        exit
```

```
      enter member-port Ethernet1/2
        exit
      exit
    enter port-channel 2
      set port-type data
      enable
      enter member-port Ethernet1/3
        exit
      enter member-port Ethernet1/4
        exit
      exit
    enter port-channel 3
      set port-type data
      enable
      enter member-port Ethernet1/5
        exit
      enter member-port Ethernet1/6
        exit
      exit
    enter port-channel 4
      set port-type mgmt
      enable
      enter member-port Ethernet2/1
        exit
      enter member-port Ethernet2/2
        exit
      exit
    enter port-channel 48
      set port-type cluster
      enable
      enter member-port Ethernet2/3
        exit
      exit
    exit
  exit
commit-buffer

scope ssa
  enter logical-device ASA1 asa "1,2,3" clustered
    enter cluster-bootstrap
      set chassis-id 1
      set ipv4 gateway 10.1.1.254
      set ipv4 pool 10.1.1.11 10.1.1.27
      set ipv6 gateway 2001:DB8::AA
      set ipv6 pool 2001:DB8::11 2001:DB8::27
      set key
      Key: f@arscape
      set mode spanned-etherchannel
      set service-type cluster1
      set virtual ipv4 10.1.1.1 mask 255.255.255.0
      set virtual ipv6 2001:DB8::1 prefix-length 64
      exit
    exit
  scope app asa 9.5.2.1
    set-default
    exit
  commit-buffer
```

For chassis 2:

```
scope eth-uplink
  scope fabric a
    create port-channel 1
```

```
      set port-type data
      enable
      create member-port Ethernet1/1
        exit
      create member-port Ethernet1/2
        exit
      exit
    create port-channel 2
      set port-type data
      enable
      create member-port Ethernet1/3
        exit
      create member-port Ethernet1/4
        exit
      exit
    create port-channel 3
      set port-type data
      enable
      create member-port Ethernet1/5
        exit
      create member-port Ethernet1/6
        exit
      exit
    create port-channel 4
      set port-type mgmt
      enable
      create member-port Ethernet2/1
        exit
      create member-port Ethernet2/2
        exit
      exit
    create port-channel 48
      set port-type cluster
      enable
      create member-port Ethernet2/3
        exit
      exit
    exit
  exit
commit-buffer

scope ssa
  enter logical-device ASA1 asa "1,2,3" clustered
    enter cluster-bootstrap
      set chassis-id 2
      set ipv4 gateway 10.1.1.254
      set ipv4 pool 10.1.1.11 10.1.1.15
      set ipv6 gateway 2001:DB8::AA
      set ipv6 pool 2001:DB8::11 2001:DB8::19
      set key
      Key: f@rscape
      set mode spanned-etherchannel
      set service-type cluster1
      set virtual ipv4 10.1.1.1 mask 255.255.255.0
      set virtual ipv6 2001:DB8::1 prefix-length 64
      exit
    exit
  scope app asa 9.5.2.1
    set-default
    exit
  commit-buffer
```

Add More Cluster Members

Add or replace an ASA cluster member.

Note This procedure only applies to adding or replacing a *chassis*; if you are adding or replacing a module to a Firepower 9300 where clustering is already enabled, the module will be added automatically.

Before you begin

- Make sure your existing cluster has enough IP addresses in the management IP address pool for this new member. If not, you need to edit the existing cluster bootstrap configuration on each chassis before you add this new member. This change causes a restart of the logical device.
- The interface configuration must be the same on the new chassis. You can export and import FXOS chassis configuration to make this process easier.
- For multiple context mode, enable multiple context mode in the ASA application on the first cluster member; additional cluster members will inherit the multiple context mode configuration automatically.

Procedure

Step 1 lick **OK**.

Step 2 To add another chassis to the cluster, repeat the procedure in Create an ASA Cluster, on page 141 except you must configure a unique **chassis-id**; otherwise, use the same configuration for the new chassis.

Add a Firepower Threat Defense Cluster

You can add a single Firepower 9300 chassis as an intra-chassis cluster.

Create a Firepower Threat Defense Cluster

You can easily deploy the cluster from the Firepower 4100/9300 chassis supervisor. All initial configuration is automatically generated for each unit.

In a Firepower 9300 chassis, you must enable clustering for all 3 module slots, even if you do not have a module installed. If you do not configure all 3 modules, the cluster will not come up.

Before you begin

- Download the application image you want to use for the logical device from Cisco.com, and then upload that image to the Firepower 4100/9300 chassis.
- Gather the following information:
  - Management interface ID, IP addresses, and network mask
  - Gateway IP address

- FMC IP address and/or NAT ID of your choosing
- DNS server IP address
- FTD hostname and domain name

Procedure

Step 1 Configure interfaces.

a) Add at least one Data type interface or EtherChannel (also known as a port-channel) before you deploy the cluster. See Add an EtherChannel (Port Channel), on page 109 or Configure a Physical Interface, on page 107.

For inter-chassis clustering, all data interfaces must be Spanned EtherChannels with at least one member interface. Add the same EtherChannels on each chassis. Combine the member interfaces from all cluster units into a single EtherChannel on the switch. See Clustering Guidelines and Limitations, on page 119 for more information about EtherChannels for inter-chassis clustering.

b) Add a Management type interface or EtherChannel. See Add an EtherChannel (Port Channel), on page 109 or Configure a Physical Interface, on page 107.

The management interface is required. Note that this management interface is not the same as the chassis management interface that is used only for chassis management (in FXOS, you might see the chassis management interface displayed as MGMT, management0, or other similar names).

For inter-chassis clustering, add the same Management interface on each chassis.

c) (Optional) Add a Firepower-eventing interface. See Add an EtherChannel (Port Channel), on page 109 or Configure a Physical Interface, on page 107.

This interface is a secondary management interface for FTD devices. To use this interface, you must configure its IP address and other parameters at the FTD CLI. For example, you can separate management traffic from events (such as web events). See the **configure network** commands in the Firepower Threat Defense command reference.

Step 2 Enter security services mode.

scope ssa

Example:

```
Firepower# scope ssa
Firepower /ssa #
```

Step 3 Set the default image version.

a) View available images. Note the Version number that you want to use.

show app

Example:

```
Firepower /ssa # show app
    Name       Version         Author     Supported Deploy Types CSP Type    Is Default
 App
```

```
    ---------- --------------- ---------- ---------------------- -----------
--------------
    asa        9.9.1           cisco      Native                 Application No
    asa        9.10.1          cisco      Native                 Application Yes
    ftd        6.2.3           cisco      Native                 Application Yes
```

b) Set the scope to the image version.

scope app ftd *application_version*

Example:

```
Firepower /ssa # scope app ftd 6.2.3
Firepower /ssa/app #
```

c) Accept the license agreement.

accept-license-agreement

Example:

```
Firepower /ssa/app # accept-license-agreement

End User License Agreement: End User License Agreement

Effective: May 22, 2017

This is an agreement between You and Cisco Systems, Inc. or its affiliates
("Cisco") and governs your Use of Cisco Software. "You" and "Your" means the
individual or legal entity licensing the Software under this EULA. "Use" or
"Using" means to download, install, activate, access or otherwise use the
Software. "Software" means the Cisco computer programs and any Upgrades made
available to You by an Approved Source and licensed to You by Cisco.
"Documentation" is the Cisco user or technical manuals, training materials,
specifications or other documentation applicable to the Software and made
available to You by an Approved Source. "Approved Source" means (i) Cisco or
(ii) the Cisco authorized reseller, distributor or systems integrator from whom
you acquired the Software. "Entitlement" means the license detail; including
license metric, duration, and quantity provided in a product ID (PID) published
on Cisco's price list, claim certificate or right to use notification.
"Upgrades" means all updates, upgrades, bug fixes, error corrections,
enhancements and other modifications to the Software and backup copies thereof.

[...]

Please "commit-buffer" if you accept the license agreement, otherwise "discard-buffer".

Firepower /ssa/app* #
```

d) Set this version as the default.

set-default

Example:

```
Firepower /ssa/app # set-default
Firepower /ssa/app* #
```

e) Save the configuration.

commit-buffer

Example:

```
Firepower /ssa/app* # commit-buffer
Firepower /ssa/app #
```

f) Exit to ssa mode.

exit

Example:

```
Firepower /ssa/app* # exit
Firepower /ssa* #
```

Example:

```
Firepower /ssa # scope app ftd 6.3.0.21
Firepower /ssa/app # set-default
Firepower /ssa/app* # accept-license-agreement
Firepower /ssa/app* # exit
Firepower /ssa* #
```

Step 4 Create the cluster:

enter logical-device *device_name* **ftd** *slots* **clustered**

- *device_name*—Used by the Firepower 4100/9300 chassis supervisor to configure clustering settings and assign interfaces; it is not the cluster name used in the security module configuration.
- *slots*—Assigns the chassis modules to the cluster. For the Firepower 4100, specify **1**. For the Firepower 9300, specify **1,2,3**. You must enable clustering for all 3 module slots in a Firepower 9300 chassis, even if you do not have a module installed. If you do not configure all 3 modules, the cluster will not come up.

Example:

```
Firepower /ssa # enter logical-device FTD1 ftd 1,2,3 clustered
Firepower /ssa/logical-device* #
```

Step 5 Configure the cluster bootstrap parameters.

These settings are meant for initial deployment only, or for disaster recovery. For normal operation, you can later change most values in the application CLI configuration.

a) Create the cluster bootstrap object.

enter cluster-bootstrap

Example:

```
Firepower /ssa/logical-device* # enter cluster-bootstrap
Firepower /ssa/logical-device/cluster-bootstrap* #
```

b) Configure an authentication key for control traffic on the cluster control link.

set key

Example:

```
Firepower /ssa/logical-device/cluster-bootstrap* # set key
Key: diamonddogs
```

You are prompted to enter the shared secret.

The shared secret is an ASCII string from 1 to 63 characters. The shared secret is used to generate the key. This option does not affect datapath traffic, including connection state update and forwarded packets, which are always sent in the clear.

c) Set the cluster interface mode.

set mode spanned-etherchannel

Spanned EtherChannel mode is the only supported mode.

Example:

```
Firepower /ssa/logical-device/cluster-bootstrap* # set mode spanned-etherchannel
Firepower /ssa/logical-device/cluster-bootstrap* #
```

d) Set the cluster group name in the security module configuration.

set service-type *cluster_name*

The name must be an ASCII string from 1 to 38 characters.

Example:

```
Firepower /ssa/logical-device/cluster-bootstrap* # set service-type cluster1
Firepower /ssa/logical-device/cluster-bootstrap* #
```

e) Exit the cluster bootstrap mode.

exit

Example:

```
Firepower /ssa/logical-device* # enter cluster-bootstrap
Firepower /ssa/logical-device/cluster-bootstrap* # set chassis-id 1
Firepower /ssa/logical-device/cluster-bootstrap* # set key
  Key: f@arscape
Firepower /ssa/logical-device/cluster-bootstrap* # set mode spanned-etherchannel
Firepower /ssa/logical-device/cluster-bootstrap* # set service-type cluster1
Firepower /ssa/logical-device/cluster-bootstrap* # exit
Firepower /ssa/logical-device/* #
```

Step 6 Configure the management bootstrap parameters.

These settings are meant for initial deployment only, or for disaster recovery. For normal operation, you can later change most values in the application CLI configuration.

a) Create the management bootstrap object.

enter mgmt-bootstrap ftd

Example:

```
Firepower /ssa/logical-device* # enter mgmt-bootstrap ftd
Firepower /ssa/logical-device/mgmt-bootstrap* #
```

b) Specify the IP address or hostname of the managing Firepower Management Center.

Set one of the following:

- **enter bootstrap-key FIREPOWER_MANAGER_IP**

  set value *IP_address*

  exit

- **enter bootstrap-key FQDN**

  set value *fmc_hostname*

  exit

c) Specify the firewall mode, routed or transparent.

create bootstrap-key FIREWALL_MODE

set value {**routed** | **transparent**}

exit

In routed mode, the device is considered to be a router hop in the network. Each interface that you want to route between is on a different subnet. A transparent firewall, on the other hand, is a Layer 2 firewall that acts like a "bump in the wire," or a "stealth firewall," and is not seen as a router hop to connected devices.

The firewall mode is only set at initial deployment. If you re-apply the bootstrap settings, this setting is not used.

Example:

```
Firepower /ssa/logical-device/mgmt-bootstrap* # create bootstrap-key FIREWALL_MODE
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # set value routed
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* #
```

d) Specify the key to be shared between the device and the FMC.

enter bootstrap-key-secret REGISTRATION_KEY

set value

Enter a value: *registration_key*

Confirm the value: *registration_key*

exit

You can choose any text string for this key between 1 and 37 characters; you will enter the same key on the FMC when you add the FTD.

Example:

```
Firepower /ssa/logical-device/mgmt-bootstrap* # create bootstrap-key-secret
REGISTRATION_KEY
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key-secret* # set value
Enter a value: gratuitousapples
Confirm the value: gratuitousapples
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key-secret* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* #
```

e) Specify a password for the FTD admin user for CLI access.

enter bootstrap-key-secret PASSWORD

set value

Enter a value: *password*

Confirm the value: *password*

exit

Example:

```
Firepower /ssa/logical-device/mgmt-bootstrap* # create bootstrap-key-secret PASSWORD
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key-secret* # set value
Enter a value: floppylampshade
Confirm the value: floppylampshade
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key-secret* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* #
```

f) Specify the fully qualified hostname.

enter bootstrap-key FQDN

set value *fqdn*

exit

Valid characters are the letters from a to z, the digits from 0 to 9, the dot (.), and the hyphen (-); maximum number of characters is 253.

Example:

```
Firepower /ssa/logical-device/mgmt-bootstrap* # create bootstrap-key FQDN
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # set value
ftdcluster1.example.com
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* #
```

g) Specify a comma-separated list of DNS servers.

enter bootstrap-key DNS_SERVERS

set value *dns_servers*

exit

The FTD uses DNS if you specify a hostname for the FMC, for example.

Example:

```
Firepower /ssa/logical-device/mgmt-bootstrap* # create bootstrap-key DNS_SERVERS
```

```
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # set value 10.9.8.7,10.9.6.5
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* #
```

h) Specify a comma-separated list of search domains.

enter bootstrap-key SEARCH_DOMAINS

set value *search_domains*

exit

Example:

```
Firepower /ssa/logical-device/mgmt-bootstrap* # create bootstrap-key SEARCH_DOMAINS
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # set value
cisco.com,example.com
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* #
```

i) Configure the management IP addresses for each security module in the cluster.

Note For the Firepower 9300, you must set the IP address for all 3 module slots in a chassis, even if you do not have a module installed. If you do not configure all 3 modules, the cluster will not come up.

To create an IPv4 management interface object:

1. Create the management interface object.

   enter ipv4 *slot_id* **firepower**

2. Set the gateway address.

   set gateway *gateway_address*

3. Set the IP address and mask.

   set ip *ip_address* **mask** *network_mask*

4. Exit the management IP mode.

   exit

5. Repeat for the remaining modules in the chassis.

To create an IPv6 management interface object:

1. Create the management interface object.

   enter ipv6 *slot_id* **firepower**

2. Set the gateway address.

   set gateway *gateway_address*

3. Set the IP address and prefix.

   set ip *ip_address* **prefix-length** *prefix*

4. Exit the management IP mode.

exit

5. Repeat for the remaining modules in the chassis.

Example:

```
Firepower /ssa/logical-device/mgmt-bootstrap* # create ipv4 1 firepower
Firepower /ssa/logical-device/mgmt-bootstrap/ipv4* # set ip 10.10.10.34 mask 255.255.255.0
Firepower /ssa/logical-device/mgmt-bootstrap/ipv4* # set gateway 10.10.10.1
Firepower /ssa/logical-device/mgmt-bootstrap/ipv4* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* # create ipv4 2 firepower
Firepower /ssa/logical-device/mgmt-bootstrap/ipv4* # set ip 10.10.10.35 mask 255.255.255.0
Firepower /ssa/logical-device/mgmt-bootstrap/ipv4* # set gateway 10.10.10.1
Firepower /ssa/logical-device/mgmt-bootstrap/ipv4* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* # create ipv4 3 firepower
Firepower /ssa/logical-device/mgmt-bootstrap/ipv4* # set ip 10.10.10.36 mask 255.255.255.0
Firepower /ssa/logical-device/mgmt-bootstrap/ipv4* # set gateway 10.10.10.1
Firepower /ssa/logical-device/mgmt-bootstrap/ipv4* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* # create ipv6 1 firepower
Firepower /ssa/logical-device/mgmt-bootstrap/ipv6* # set ip 2001:0DB8:BA98::3210
prefix-length 64
Firepower /ssa/logical-device/mgmt-bootstrap/ipv6* # set gateway 2001:0DB8:BA98::3211
Firepower /ssa/logical-device/mgmt-bootstrap/ipv6* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* # create ipv6 1 firepower
Firepower /ssa/logical-device/mgmt-bootstrap/ipv6* # set ip 2001:0DB8:BA98::3210
prefix-length 64
Firepower /ssa/logical-device/mgmt-bootstrap/ipv6* # set gateway 2001:0DB8:BA98::3211
Firepower /ssa/logical-device/mgmt-bootstrap/ipv6* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* # create ipv6 2 firepower
Firepower /ssa/logical-device/mgmt-bootstrap/ipv6* # set ip 2001:0DB8:BA98::3211
prefix-length 64
Firepower /ssa/logical-device/mgmt-bootstrap/ipv6* # set gateway 2001:0DB8:BA98::3211
Firepower /ssa/logical-device/mgmt-bootstrap/ipv6* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* # create ipv6 3 firepower
Firepower /ssa/logical-device/mgmt-bootstrap/ipv6* # set ip 2001:0DB8:BA98::3212
prefix-length 64
Firepower /ssa/logical-device/mgmt-bootstrap/ipv6* # set gateway 2001:0DB8:BA98::3211
Firepower /ssa/logical-device/mgmt-bootstrap/ipv6* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* #
```

j) Exit the management bootstrap mode.

exit

Example:

```
Firepower /ssa/logical-device/mgmt-bootstrap* # exit
Firepower /ssa/logical-device* #
```

Example:

```
Firepower /ssa/logical-device* # enter mgmt-bootstrap ftd
Firepower /ssa/logical-device/mgmt-bootstrap* # enter bootstrap-key FIREPOWER_MANAGER_IP
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # set value 10.0.0.100
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* # enter bootstrap-key FIREWALL_MODE
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # set value routed
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* # enter bootstrap-key-secret REGISTRATION_KEY
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key-secret* # set value
```

```
  Value: ziggy$tardust
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key-secret* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* # enter bootstrap-key-secret PASSWORD
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key-secret* # set value
  Value: $pidersfrommars
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key-secret* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* # enter bootstrap-key FQDN
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # set value example.cisco.com
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* # enter bootstrap-key DNS_SERVERS
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # set value 192.168.1.1
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* # enter bootstrap-key SEARCH_DOMAINS
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # set value example.com
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* # enter ipv4 1 firepower
Firepower /ssa/logical-device/mgmt-bootstrap/ipv4* # set gateway 10.0.0.1
Firepower /ssa/logical-device/mgmt-bootstrap/ipv4* # set ip 10.0.0.31 mask 255.255.255.0
Firepower /ssa/logical-device/mgmt-bootstrap/ipv4* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* # enter ipv4 2 firepower
Firepower /ssa/logical-device/mgmt-bootstrap/ipv4* # set gateway 10.0.0.1
Firepower /ssa/logical-device/mgmt-bootstrap/ipv4* # set ip 10.0.0.32 mask 255.255.255.0
Firepower /ssa/logical-device/mgmt-bootstrap/ipv4* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* # enter ipv4 3 firepower
Firepower /ssa/logical-device/mgmt-bootstrap/ipv4* # set gateway 10.0.0.1
Firepower /ssa/logical-device/mgmt-bootstrap/ipv4* # set ip 10.0.0.33 mask 255.255.255.0
Firepower /ssa/logical-device/mgmt-bootstrap/ipv4* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* # exit
Firepower /ssa/logical-device* #
```

Step 7 Save the configuration.

commit-buffer

The chassis deploys the logical device by downloading the specified software version and pushing the bootstrap configuration and management interface settings to the application instance. Check the status of the deployment using the **show app-instance** command. The application instance is running and ready to use when the **Admin State** is **Enabled** and the **Oper State** is **Online**.

Example:

```
Firepower /ssa/logical-device* # commit-buffer
Firepower /ssa/logical-device # exit
Firepower /ssa # show app-instance
App Name   Identifier Slot ID    Admin State Oper State      Running Version Startup Version
 Deploy Type Profile Name Cluster State   Cluster Role
---------- ---------- ---------- ----------- ---------------- --------------- ---------------
 ----------- ------------ --------------- -------------
ftd        cluster1   1          Enabled     Online           6.4.0.49        6.4.0.49
    Native                    In Cluster      Slave
ftd        cluster1   2          Enabled     Online           6.4.0.49        6.4.0.49
    Native                    In Cluster      Master
ftd        cluster1   3          Disabled    Not Available                    6.4.0.49
    Native                    Not Applicable  None
```

Step 8 Add each unit to the Firepower Management Center using the management IP addresses, and then group them into a cluster at the web interface.

All cluster units must be in a successfully-formed cluster on FXOS prior to adding them to Firepower Management Center.

Example

```
scope eth-uplink
  scope fabric a
    enter port-channel 1
      set port-type data
      enable
      create member-port Ethernet1/1
        exit
      create member-port Ethernet1/2
        exit
      exit
    enter port-channel 2
      set port-type data
      enable
      create member-port Ethernet1/3
        exit
      create member-port Ethernet1/4
        exit
      exit
    enter port-channel 3
      set port-type firepower-eventing
      enable
      create member-port Ethernet1/5
        exit
      create member-port Ethernet1/6
        exit
      exit
    enter port-channel 4
      set port-type mgmt
      enable
      create member-port Ethernet2/1
        exit
      enter member-port Ethernet2/2
        exit
      exit

    exit
  exit
commit-buffer

scope ssa
  enter logical-device FTD1 ftd "1,2,3" clustered
    enter cluster-bootstrap
      set chassis-id 1
      set key cluster_key
      set mode spanned-etherchannel
      set service-type ftd-cluster
      exit
    enter mgmt-bootstrap ftd
      enter bootstrap-key FIREPOWER_MANAGER_IP
        set value 10.0.0.100
        exit
      enter bootstrap-key FIREWALL_MODE
        set value transparent
        exit
```

```
    enter bootstrap-key-secret REGISTRATION_KEY
      set value
        Value: alladinsane
      exit
    enter bootstrap-key-secret PASSWORD
      set value
        Value: widthofacircle
      exit
    enter bootstrap-key FQDN
      set value ftd.cisco.com
      exit
    enter bootstrap-key DNS_SERVERS
      set value 192.168.1.1
      exit
    enter bootstrap-key SEARCH_DOMAINS
      set value search.com
      exit
    enter ipv4 1 firepower
      set gateway 10.0.0.1
      set ip 10.0.0.31 mask 255.255.255.0
      exit
    enter ipv4 2 firepower
      set gateway 10.0.0.1
      set ip 10.0.0.32 mask 255.255.255.0
      exit
    enter ipv4 3 firepower
      set gateway 10.0.0.1
      set ip 10.0.0.33 mask 255.255.255.0
      exit
    exit
  exit
scope app ftd 6.0.0.837
  accept-license-agreement
  set-default
  exit
commit-buffer
```

Add More Cluster Units

Add or replace a FTD cluster unit in an existing cluster.

Note The FXOS steps in this procedure only apply to adding a new *chassis*; if you are adding a new module to a Firepower 9300 where clustering is already enabled, the module will be added automatically. However, you must still add the new module to the Firepower Management Center; skip to the Firepower Management Center steps.

Before you begin

- In the case of a replacement, you must delete the old cluster unit from the Firepower Management Center. When you replace it with a new unit, it is considered to be a new device on the Firepower Management Center.
- The interface configuration must be the same on the new chassis. You can export and import FXOS chassis configuration to make this process easier.

Procedure

To add another chassis to the cluster, repeat the procedure in Create a Firepower Threat Defense Cluster, on page 149 except you must configure the following settings to be unique; otherwise, use the same configuration for both chassis.

- Chassis ID
- Management IP addresses

Configure Radware DefensePro

The Cisco Firepower 4100/9300 chassis can support multiple services (for example, a firewall and a third-party DDoS application) on a single blade. These applications and services can be linked together to form a Service Chain.

About Radware DefensePro

In the current supported Service Chaining configuration, the third-party Radware DefensePro virtual platform runs in front of the ASA firewall. Radware DefensePro is a KVM-based virtual platform that provides distributed denial-of-service (DDoS) detection and mitigation capabilities on the Firepower 4100/9300 chassis. When Service Chaining is enabled on your Firepower 4100/9300 chassis, traffic from the network must first pass through the DefensePro virtual platform before reaching the main ASA firewall.

Note

- The Radware DefensePro virtual platform may be referred to as *Radware vDP* (virtual DefensePro), or simply *vDP*.
- The Radware DefensePro virtual platform may occasionally be referred to as a Link Decorator.

Prerequisites for Radware DefensePro

Prior to deploying Radware DefensePro on your Firepower 4100/9300 chassis, you must configure the Firepower 4100/9300 chassis to use an NTP Server with the **etc/UTC** Time Zone. For more information about setting the date and time in your Firepower 4100/9300 chassis, see Setting the Date and Time, on page 57.

Guidelines for Service Chaining

Models

- ASA—The Radware DefensePro (vDP) platform is supported with ASA on the following models:
  - Firepower 9300

- Firepower 4120—You must use the CLI to deploy Radware DefensePro on this platform; the Firepower Chassis Manager does not yet support this functionality.
- Firepower 4140—You must use the CLI to deploy Radware DefensePro on this platform; the Firepower Chassis Manager does not yet support this functionality.
- Firepower 4150

Note The Radware DefensePro platform is not currently supported with ASA on Firepower 4110 devices.

Additional Guidelines

Configure Radware DefensePro on a Standalone Logical Device

The following procedure shows how to install Radware DefensePro in a single Service Chain in front of a standalone ASA logical device.

Before you begin

- Download the vDP image from Cisco.com (see Downloading Images from Cisco.com, on page 46) and then download that image to the Firepower 4100/9300 chassis (see Downloading a Logical Device Software Image to the Firepower 4100/9300 chassis, on page 49).
- You can deploy the Radware DefensePro application in a standalone configuration on an intra-chassis cluster; for intra-chassis clustering, see Configure Radware DefensePro on an Intra-Chassis Cluster, on page 164.

Procedure

Step 1 If you want to use a separate management interface for vDP, enable the interface and set it to be the mgmt type according to Configure a Physical Interface, on page 107. Otherwise, you can share the application management interface.

Step 2 Create an ASA logical device in standalone configuration (see Add a Standalone ASA, on page 123).

Step 3 Enter security services mode:

Firepower# **scope ssa**

Step 4 Create the Radware vDP instance:

Firepower /ssa # **scope slot** *slot_id*

Firepower /ssa/slot # **create app-instance vdp** *logical_device_identifier*

Firepower /ssa/slot/app-instance* # **exit**

Firepower /ssa/slot/* # **exit**

Step 5 Commit the configuration:

commit-buffer

Step 6 Verify the installation and provisioning of vDP on the security module:

Firepower /ssa # **show app-instance**

Step 7 Once the vDP application is installed, access the logical device:

Firepower /ssa # **scope logical-device** *device_name*

Step 8 Assign the management interface to vDP. You can use the same physical interface as for the logical device, or you can use a separate interface.

Firepower /ssa/logical-device # **enter external-port-link** *name interface_id* **vdp**

Firepower /ssa/logical-device/external-port-link* # **exit**

Step 9 Configure the external management interface settings for vDP.

a) Create the bootstrap object:

Firepower /ssa/logical-device* # **create mgmt-bootstrap vdp**

b) Configure the management IP address:

Firepower /ssa/logical-device/mgmt-bootstrap* #**create ipv4** *slot_id* **default**

c) Set the gateway address:

Firepower /ssa/logical-device/mgmt-bootstrap/ipv4* #**set gateway** *gateway_address*

d) Set the IP address and mask:

Firepower /ssa/logical-device/mgmt-bootstrap/ipv4* #**set ip** *ip_address* **mask** *network_mask*

e) Exit the management IP configuration scope:

Firepower /ssa/logical-device/mgmt-bootstrap/ipv4* #**exit**

f) Exit the management bootstrap configuration scope:

Firepower /ssa/logical-device/mgmt-bootstrap* #**exit**

Step 10 Edit the data interface where you want to place the vDP in front of the ASA flow:

Firepower /ssa/logical-device* # **scope external-port-link** *name*

Enter the **show external-port-link** command to view interface names.

Step 11 Add the vDP to the logical device:

Firepower /ssa/logical-device/external-port-link* # **set decorator vdp**

Repeat for each interface where you want to use vDP.

Step 12 Commit the configuration:

commit-buffer

Step 13 Verify that the third-party app is set for the interface:

Firepower /ssa/logical-device/external-port-link* # **show detail**

What to do next

Set a password for the DefensePro application. Note that the application does not come online until you set a password. For more information, see the Radware DefensePro DDoS Mitigation User Guide on cisco.com.

Configure Radware DefensePro on an Intra-Chassis Cluster

Note Service Chaining is not supported in an inter-chassis cluster configuration. However, the Radware DefensePro application can be deployed in a standalone configuration in an inter-chassis cluster scenario.

Before you begin

- Download the vDP image from Cisco.com (see Downloading Images from Cisco.com, on page 46) and then download that image to the Firepower 4100/9300 chassis (see Downloading a Logical Device Software Image to the Firepower 4100/9300 chassis, on page 49).

Procedure

Step 1 If you want to use a separate management interface for vDP, enable the interface and set it to be the mgmt type according to Configure a Physical Interface, on page 107. Otherwise, you can share the application management interface.

Step 2 Configure an ASA intra-chassis cluster (see Create an ASA Cluster, on page 141)

Step 3 Decorate the external (client-facing) port with Radware DefensePro:

enter external-port-link *name interface_name* { **asa** }

set **decorator vdp**

set **description** ''''

exit

Step 4 Assign the external management port for the logical device:

enter **external-port-link** *{ mgmt_asa | mgmt_ftd } interface_id { asa | ftd }*

set **decorator** ''''

set **description** ''''

exit

Step 5 Assign the external management port for DefensePro:

enter **external-port-link mgmt_vdp** *interface_name { asa | ftd }*

set **decorator** ''''

set **description** ''''

Step 6 Configure cluster port channel:

enter **external-port-link** port-channel48 Port-channel48 *{ asa | ftd }*

set **decorator** ""

set **description** ""

exit

Step 7 Configure management bootstrap for all three DefensePro instances:

enter **mgmt-bootstrap vdp**

enter **ipv4** *slot_id* **default**

set gateway *gateway_address*

set ip *ip_address* **mask** *network_mask*

exit

Example:

```
enter mgmt-bootstrap vdp
    enter ipv4 1 default
        set gateway 172.16.0.1
        set ip 172.16.4.219 mask 255.255.0.0
    exit

    enter ipv4 2 default
        set gateway 172.16.0.1
        set ip 172.16.4.220 mask 255.255.0.0
    exit

    enter ipv4 3 default
        set gateway 172.16.0.1
        set ip 172.16.4.221 mask 255.255.0.0
    exit
```

Step 8 Exit management bootstrap configuration scope:

exit

Step 9 Enter the DefensePro application instance on the Control blade:

connect module *slot* **console**

connect vdp

Step 10 On the Control blade, set the management IP:

device clustering management-channel ip

Step 11 Using the IP found in the previous step, set the Control IP:

device clustering master set *management-channel ip*

Step 12 Enable the cluster:

device clustering state set enable

Step 13 Exit the application console and return to the FXOS module CLI:

Ctrl]

Step 14 Repeat steps 10, 12, 13, and 14 to set the Control blade IP found in step 11 and enable the cluster for each blade application instance.

Step 15 Commit the configuration:

commit-buffer

Note After completing this procedure, you must verify whether the DefensePro instances are configured in a cluster.

Step 16 Validate that all DefensePro applications have joined the cluster:

device cluster show

Step 17 Use either of the following methods to verify which DefensePro instance is primary, and which one is secondary.

a) Scope the DefensePro instance and show appplication attributes for DefensePro only:

scope ssa

scope slot *slot_number*

scope app-instance vdp

show app-attri

b) Scope the slot and show the DefensePro instance in expanded detail. This approach displays information for both logical device and vDP application instances on the slot.

scope ssa

scope *slot_number*

show app-instance expand detail

If the DefensePro application is online but not yet formed in a cluster, the CLI displays:

```
App Attribute:
    App Attribute Key: cluster-role
    Value: unknown
```

If the system displays this "unknown" value, you must enter the DefensePro application and configure the Control blade IP address to create the vDP cluster.

If the DefensePro application is online and formed in a cluster, the CLI displays:

```
App Attribute:
    App Attribute Key: cluster-role
    Value: primary/secondary
```

Example

```
scope ssa
  enter logical-device ld asa "1,2,3" clustered
     enter cluster-bootstrap
         set chassis-id 1
         set ipv4 gateway 172.16.0.1
         set ipv4 pool 172.16.4.216 172.16.4.218
         set ipv6 gateway 2010::2
         set ipv6 pool 2010::21 2010::26
         set key secret
         set mode spanned-etherchannel
         set name cisco
         set virtual ipv4 172.16.4.222 mask 255.255.0.0
```

```
            set virtual ipv6 2010::134 prefix-length 64
        exit
        enter external-port-link Ethernet1-2 Ethernet1/2 asa
            set decorator vdp
            set description ""
        exit
        enter external-port-link Ethernet1-3_asa Ethernet1/3 asa
            set decorator ""
            set description ""
        exit
        enter external-port-link mgmt_asa Ethernet1/1 asa
            set decorator ""
            set description ""
        exit
        enter external-port-link mgmt_vdp Ethernet1/1 vdp
            set decorator ""
            set description ""
        exit
        enter external-port-link port-channel48 Port-channel48 asa
            set decorator ""
            set description ""
        exit
        enter mgmt-bootstrap vdp
            enter ipv4 1 default
                set gateway 172.16.0.1
                set ip 172.16.4.219 mask 255.255.0.0
            exit

            enter ipv4 2 default
                set gateway 172.16.0.1
                set ip 172.16.4.220 mask 255.255.0.0
            exit

            enter ipv4 3 default
                set gateway 172.16.0.1
                set ip 172.16.4.221 mask 255.255.0.0
            exit
 exit
commit-buffer
scope ssa
   scope slot 1
   scope app-instance vdp
   show app-attri
    App Attribute:
        App Attribute Key: cluster-role
        Value: unknown
```

What to do next

Set a password for the DefensePro application. Note that the application does not come online until you set a password. For more information, see the Radware DefensePro DDoS Mitigation User Guide on cisco.com.

Open UDP/TCP Ports and Enable vDP Web Services

The Radware APSolute Vision Manager interfaces communicate with the Radware vDP appliation using various UDP/TCP ports. In orer for the vDP application to communicate with the APSolute Vision Manager, you must ensure that these ports are accessible and not blocked by your firewall. For more information on which specific ports to open, see the following tables in the APSolute Vision User Guide:

- **Ports for APSolute Vision Server-WBM Communication and Operating System**
- **Communication Ports for APSolute Vision Server with Radware Devices**

In order for Radware APSolute Vision to manage the Virtual DefensePro application deployed on the FXOS chassis, you must enable the vDP web service using the FXOS CLI.

Procedure

Step 1 From the FXOS CLI, connect to the vDP application instance.

connect module *slot* **console**

connect vdp

Step 2 Enable vDP web services.

manage secure-web status set enable

Step 3 Exit the vDP application console and return to the FXOS module CLI.

Ctrl]

Manage Logical Devices

You can delete a logical device, convert an ASA to transparent mode, change the interface configuration, and perform other tasks on existing logical devices.

Connect to the Console of the Application

Use the following procedure to connect to the console of the application.

Procedure

Step 1 Connect to the module CLI.

connect module *slot_number* **console**

To connect to the security engine of a device that does not support multiple security modules, always use **1** as the *slot_number*.

Example:

```
Firepower# connect module 1 console
Telnet escape character is '~'.
Trying 127.5.1.1...
Connected to 127.5.1.1.
Escape character is '~'.

CISCO Serial Over LAN:
Close Network Connection to Exit
```

```
Firepower-module1>
```

Step 2 Connect to the application console. Enter the appropriate command for your device.

connect ftd

connect vdp

Step 3 Exit the application console to the FXOS module CLI.

- FTD—Enter
- vDP—Enter **Ctrl-], .**

Step 4 Return to the supervisor level of the FXOS CLI.

a) Enter **~**

You exit to the Telnet application.

b) To exit the Telnet application, enter:

telnet>**quit**

Delete a Logical Device

Procedure

Step 1 Enter security services mode:

Firepower# **scope ssa**

Step 2 View details for the logical devices on the chassis:

Firepower /ssa # **show logical-device**

Step 3 For each logical device that you want to delete, enter the following command:

Firepower /ssa # **delete logical-device** *device_name*

Step 4 View details for the applications installed on the logical devices:

Firepower /ssa # **show app-instance**

Step 5 For each application that you want to delete, enter the following commands:

a) Firepower /ssa # **scope slot** *slot_number*
b) Firepower /ssa/slot # **delete app-instance** *application_name*
c) Firepower /ssa/slot # **exit**

Step 6 Commit the configuration:

commit-buffer

Commits the transaction to the system configuration.

Example

```
Firepower# scope ssa
Firepower /ssa # show logical-device

Logical Device:
    Name       Description Slot ID    Mode       Operational State        Template Name
    ---------- ----------- ---------- ---------- ------------------------ -------------
    FTD                    1,2,3      Clustered  Ok                       ftd
Firepower /ssa # delete logical-device FTD
Firepower /ssa* # show app-instance
Application Name     Slot ID     Admin State     Operational State    Running Version Startup
 Version Cluster Oper State
-------------------- ----------- --------------- -------------------- ---------------
--------------- ------------------
ftd                              1 Disabled        Stopping             6.0.0.837
6.0.0.837       Not Applicable
ftd                              2 Disabled        Offline              6.0.0.837
6.0.0.837       Not Applicable
ftd                              3 Disabled        Not Available
6.0.0.837       Not Applicable
Firepower /ssa* # scope slot 1
Firepower /ssa/slot # delete app-instance ftd
Firepower /ssa/slot* # exit
Firepower /ssa* # scope slot 2
Firepower /ssa/slot # delete app-instance ftd
Firepower /ssa/slot* # exit
Firepower /ssa* # scope slot 3
Firepower /ssa/slot # delete app-instance ftd
Firepower /ssa/slot* # exit
Firepower /ssa* # commit-buffer
```

Remove a Cluster Unit

The following sections describe how to remove units temporarily or permanently from the cluster.

Temporary Removal

A cluster unit will be automatically removed from the cluster due to a hardware or network failure, for example. This removal is temporary until the conditions are rectified, and it can rejoin the cluster. You can also manually disable clustering.

To check whether a device is currently in the cluster, check the cluster status within the application using the **show cluster info** command:

```
ciscoasa# show cluster info
Clustering is not enabled
```

For FTD using FMC, you should leave the device in the FMC device list so that it can resume full functionality after you reenable clustering.

- Disable clustering in the application—You can disable clustering using the application CLI. Enter the **cluster remove unit** *name* command to remove any unit other than the one you are logged into. The

bootstrap configuration remains intact, as well as the last configuration synced from the control unit, so you can later re-add the unit without losing your configuration. If you enter this command on a data unit to remove the control unit, a new control unit is elected.

When a device becomes inactive, all data interfaces are shut down; only the Management interface can send and receive traffic. To resume traffic flow, re-enable clustering. The Management interface remains up using the IP address the unit received from the bootstrap configuration. However if you reload, and the unit is still inactive in the cluster , the Management interface is disabled.

To reenable clustering, on the FTD enter **cluster enable**.

- Disable the application instance—At the FXOS CLI, see the following example:

```
Firepower-chassis# scope ssa
Firepower-chassis /ssa # scope slot 1
Firepower-chassis /ssa/slot # scope app-instance asa asa1
Firepower-chassis /ssa/slot/app-instance # disable
Firepower-chassis /ssa/slot/app-instance* # commit-buffer
Firepower-chassis /ssa/slot/app-instance #
```

To reenable:

```
Firepower-chassis /ssa/slot/app-instance # enable
Firepower-chassis /ssa/slot/app-instance* # commit-buffer
Firepower-chassis /ssa/slot/app-instance #
```

- Shut down the security module/engine—At the FXOS CLI, see the following example:

```
Firepower-chassis# scope service-profile server 1/1
Firepower-chassis /org/service-profile # power down soft-shut-down
Firepower-chassis /org/service-profile* # commit-buffer
Firepower-chassis /org/service-profile #
```

To power up:

```
Firepower-chassis /org/service-profile # power up
Firepower-chassis /org/service-profile* # commit-buffer
Firepower-chassis /org/service-profile #
```

- Shut down the chassis—At the FXOS CLI, see the following example:

```
Firepower-chassis# scope chassis 1
Firepower-chassis /chassis # shutdown no-prompt
```

Permanent Removal

You can permanently remove a cluster member using the following methods.

For FTD using FMC, be sure to remove the unit from the FMC device list after you disable clustering on the chassis.

- Delete the logical device—At the FXOS CLI, see the following example:

```
Firepower-chassis# scope ssa
Firepower-chassis /ssa # delete logical-device cluster1
Firepower-chassis /ssa* # commit-buffer
Firepower-chassis /ssa #
```

- Remove the chassis or security module from service—If you remove a device from service, you can add replacement hardware as a new member of the cluster.

Change the ASA to Transparent Firewall Mode

You can only deploy a routed firewall mode ASA from the Firepower 4100/9300 chassis. To change the ASA to transparent firewall mode, complete the initial deployment, and then change the firewall mode within the ASA CLI. For standalone ASAs, because changing the firewall mode erases the configuration, you must then redeploy the configuration from the Firepower 4100/9300 chassis to regain the bootstrap configuration. The ASA then remains in transparent mode with a working bootstrap configuration. For clustered ASAs, the configuration is not erased, so you do not need to redeploy the bootstrap configuration from FXOS.

Procedure

Step 1 Connect to the ASA console according to Connect to the Console of the Application, on page 168. For a cluster, connect to the primary unit. For a failover pair, connect to the active unit.

Step 2 Enter configuration mode:

enable

configure terminal

By default, the enable password is blank.

Step 3 Set the firewall mode to transparent:

firewall transparent

Step 4 Save the configuration:

write memory

For a cluster or failover pair, this configuration is replicated to secondary units:

```
asa(config)# firewall transparent
asa(config)# write memory
Building configuration...
Cryptochecksum: 9f831dfb 60dffa8c 1d939884 74735b69

3791 bytes copied in 0.160 secs
[OK]
asa(config)#
Beginning configuration replication to unit-1-2
 End Configuration Replication to data unit.

asa(config)#
```

Step 5 On the Firepower Chassis Manager **Logical Devices** page, click the **Edit** icon to edit the ASA.

The **Provisioning** page appears.

Step 6 Click the device icon to edit the bootstrap configuration. Change any value in your configuration, and click **OK**.

You must change the value of at least one field, for example, the **Password** field.

You see a warning about changing the bootstrap configuration; click **Yes**.

Step 7 For an inter-chassis cluster or for a failover pair, repeat steps 5 through 7 to redeploy the bootstrap configuration on each chassis.

Wait several minutes for the chassis/security modules to reload, and for the ASA to become operational again. The ASA now has an operational bootstrap configuration, but remains in transparent mode.

Change an Interface on a Firepower Threat Defense Logical Device

You can allocate or unallocate an interface on the FTD logical device. You can then sync the interface configuration in FMC.

Adding a new interface, or deleting an unused interface has minimal impact on the FTD configuration. However, deleting an interface that is used in your security policy will impact the configuration. Interfaces can be referenced directly in many places in the FTD configuration, including access rules, NAT, SSL, identity rules, VPN, DHCP server, and so on. Policies that refer to security zones are not affected. You can also edit the membership of an allocated EtherChannel without affecting the logical device or requiring a sync on the FMC.

Deleting an interface will delete any configuration associated with that interface.

Before you begin

- Configure your interfaces, and add any EtherChannels according to Configure a Physical Interface, on page 107 and Add an EtherChannel (Port Channel), on page 109.
- If you want to add an already-allocated interface to an EtherChannel (for example, all interfaces are allocated by default to a cluster), you need to unallocate the interface from the logical device first, then add the interface to the EtherChannel. For a new EtherChannel, you can then allocate the EtherChannel to the device.
- If you want to replace the management or firepower eventing interface, you must use the Firepower Chassis Manager; the CLI does not support this change.
- For clustering or High Availability, make sure you add or remove the interface on all units before you sync the configuration in the FMC. We recommend that you make the interface changes on the data/standby unit(s) first, and then on the control/active unit. Note that new interfaces are added in an administratively down state, so they do not affect interface monitoring.

Procedure

Step 1 Enter security services mode:

Firepower# **scope ssa**

Step 2 Edit the logical device:

Firepower /ssa # **scope logical-device** *device_name*

Step 3 Allocate a new interface to the logical device:

Firepower /ssa/logical-device* # **create external-port-link** *name interface_id* **ftd**

Do not delete any interfaces yet.

Step 4 Commit the configuration:

commit-buffer

Commits the transaction to the system configuration.

Step 5 Sync the interfaces in FMC.

a) Log into the FMC.

b) Select **Devices** > **Device Management** and click **Edit** (✎) for your FTD device. The **Interfaces** page is selected by default.

c) Click the **Sync Device** button on the top left of the **Interfaces** page.

d) After the changes are detected, you will see a red banner on the **Interfaces** page indicating that the interface configuration has changed. Click the **Click to know more** link to view the interface changes.

e) If you plan to delete an interface, manually transfer any interface configuration from the old interface to the new interface.

Because you have not yet deleted any interfaces, you can refer to the existing configuration.

f) Click **Save**.

g) Select the devices and click **Deploy** to deploy the policy to the assigned devices. The changes are not active until you deploy them.

Step 6 In FXOS, unallocate an interface from the logical device:

Firepower /ssa/logical-device # **delete external-port-link** *name*

Enter the **show external-port-link** command to view interface names.

Step 7 Commit the configuration:

commit-buffer

Commits the transaction to the system configuration.

Step 8 Sync the interfaces again in FMC.

Change an Interface on an ASA Logical Device

You can allocate, unallocate, or replace a management interface on an ASA logical device. ASDM discovers the new interfaces automatically.

Adding a new interface, or deleting an unused interface has minimal impact on the ASA configuration. However, if you remove an allocated interface in FXOS (for example, if you remove a network module, remove an EtherChannel, or reassign an allocated interface to an EtherChannel), and the interface is used in your security policy, removal will impact the ASA configuration. In this case, the ASA configuration retains

the original commands so that you can make any necessary adjustments. You can manually remove the old interface configuration in the ASA OS.

Note You can edit the membership of an allocated EtherChannel without impacting the logical device.

Before you begin

- Configure your interfaces and add any EtherChannels according to Configure a Physical Interface, on page 107 and Add an EtherChannel (Port Channel), on page 109.
- If you want to add an already-allocated interface to an EtherChannel (for example, all interfaces are allocated by default to a cluster), you need to unallocate the interface from the logical device first, then add the interface to the EtherChannel. For a new EtherChannel, you can then allocate the EtherChannel to the device.
- For clustering or failover, make sure you add or remove the interface on all units. We recommend that you make the interface changes on the data/standby unit(s) first, and then on the control/active unit. New interfaces are added in an administratively down state, so they do not affect interface monitoring.

Procedure

Step 1 Enter security services mode:

Firepower# **scope ssa**

Step 2 Edit the logical device:

Firepower /ssa # **scope logical-device** *device_name*

Step 3 Unallocate an interface from the logical device:

Firepower /ssa/logical-device # **delete external-port-link** *name*

Enter the **show external-port-link** command to view interface names.

For a management interface, delete the current interface then commit your change using the **commit-buffer** command before you add the new management interface.

Step 4 Allocate a new interface to the logical device:

Firepower /ssa/logical-device* # **create external-port-link** *name interface_id* **asa**

Step 5 Commit the configuration:

commit-buffer

Commits the transaction to the system configuration.

Monitoring Logical Devices

- **show app**

  View available images.

```
Firepower# scope ssa
Firepower /ssa # show app
    Name       Version         Author     Supported Deploy Types CSP Type    Is Default
 App
    ---------- --------------- ---------- ---------------------- -----------
--------------
    asa        9.10.1          cisco      Native                 Application Yes
    ftd        6.2.3           cisco      Native                 Application Yes
    vdp        8.13.01.09-2    radware    Vm                     Application Yes
```

- **show app-instance**

  View the application instance status and information.

```
Firepower# scope ssa
Firepower /ssa # show app-instance
App Name   Slot ID    Admin State Oper State       Running Version Startup Version
Cluster State   Cluster Role
---------- ---------- ----------- ---------------- --------------- ---------------
--------------- ------------
ftd        1          Enabled     Online           6.2.1.62        6.2.1.62        Not
 Applicable  None
vdp        1          Disabled    Installing                       8.10.01.16-5    Not
 Applicable  None
```

- **show logical-device**

  View details for logical devices.

```
Firepower# scope ssa
Firepower /ssa # show logical-device

Logical Device:
    Name       Description Slot ID    Mode       Oper State              Template Name

    ---------- ----------- ---------- ---------- ----------------------- -------------

    asa1                   1          Standalone Ok                       asa
```

- **show app-resource-profile**

  Show resource profiles for vDP.

```
Firepower# scope ssa
Firepower /ssa # scope app vdp 8.13.01.09-2
Firepower /ssa/app # show app-resource-profile
Profile Name             Security Model  CPU Logical Core Count RAM Size (MB)   Default
 Profile
------------------------ --------------- ---------------------- ---------------
---------------
```

```
DEFAULT-4110-RESOURCE      FPR4K-SM-12                              4           16384 Yes
DEFAULT-RESOURCE         FPR9K-SM-56, FPR9K-SM-44, FPR9K-SM-36, FPR9K-SM-24, FPR4K-SM-44,
 FPR4K-SM-36, FPR4K-SM-24
                                                                  6           24576 Yes
VDP-10-CORES             FPR9K-SM-56, FPR9K-SM-44, FPR9K-SM-36, FPR9K-SM-24, FPR4K-SM-44,
 FPR4K-SM-36, FPR4K-SM-24
                                                                 10           40960 No
VDP-2-CORES                all                                    2            8192 No
VDP-4-CORES                all                                    4           16384 No
VDP-8-CORES              FPR9K-SM-56, FPR9K-SM-44, FPR9K-SM-36, FPR9K-SM-24, FPR4K-SM-44,
 FPR4K-SM-36, FPR4K-SM-24
                                                                  8           32768 No
```

Examples for Inter-Site Clustering

The following examples show supported cluster deployments.

Spanned EtherChannel Routed Mode Example with Site-Specific MAC Addresses

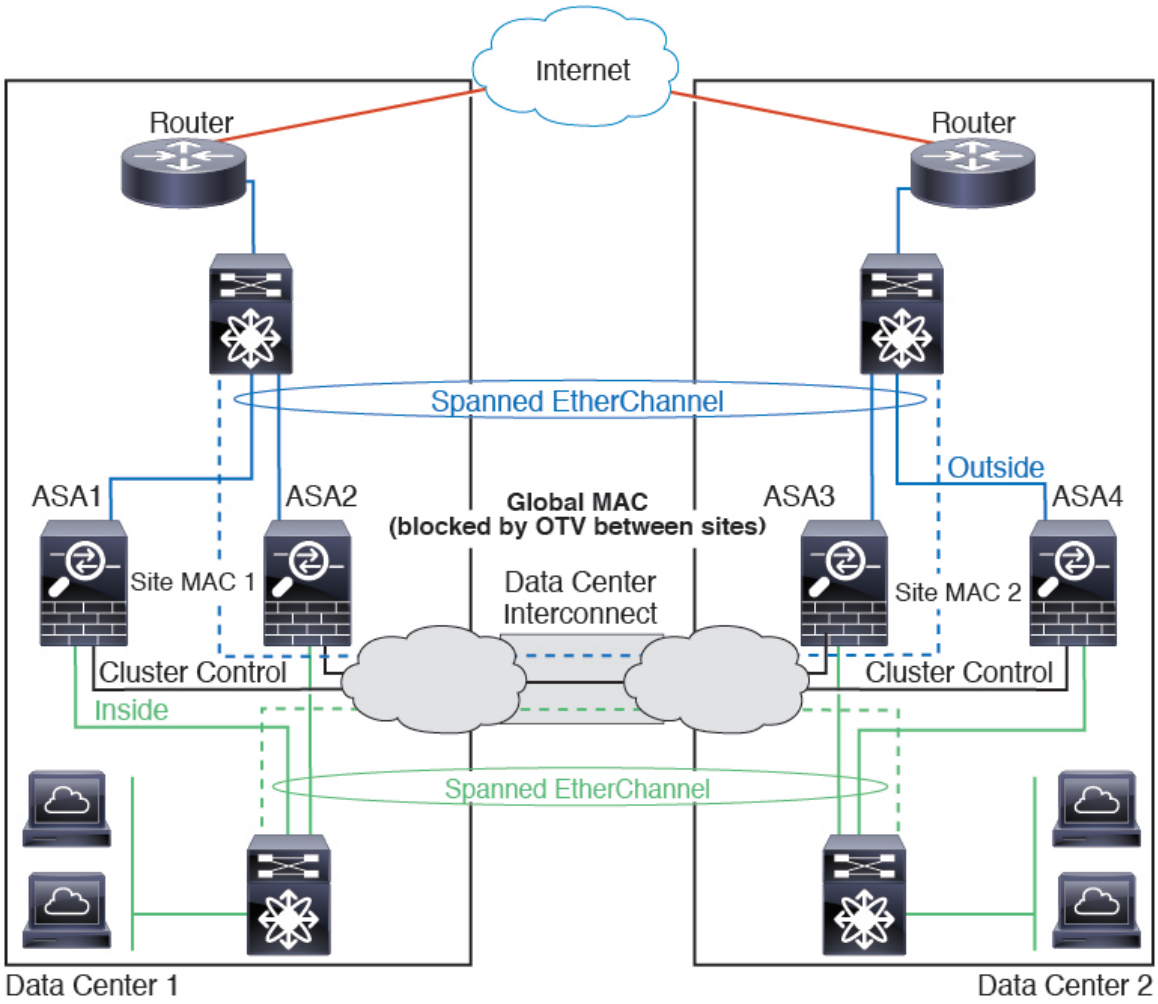
The following example shows 2 cluster members at each of 2 data centers placed between the gateway router and an inside network at each site (East-West insertion). The cluster members are connected by the cluster control link over the DCI. The cluster members at each site connect to the local switches using spanned EtherChannels for both the inside and outside networks. Each EtherChannel is spanned across all chassis in the cluster.

The data VLANs are extended between the sites using Overlay Transport Virtualization (OTV) (or something similar). You must add filters blocking the global MAC address to prevent traffic from traversing the DCI to the other site when the traffic is destined for the cluster. If the cluster units at one site become unreachable, you must remove the filters so traffic can be sent to the other site's cluster units. You should use VACLs to filter the global MAC address.Be sure to disable ARP inspection.

The cluster acts as the gateway for the inside networks. The global virtual MAC, which is shared across all cluster units, is used only to receive packets. Outgoing packets use a site-specific MAC address from each DC cluster. This feature prevents the switches from learning the same global MAC address from both sites on two different ports, which causes MAC flapping; instead, they only learn the site MAC address.

In this scenario:

- All egress packets sent from the cluster use the site MAC address and are localized at the data center.
- All ingress packets to the cluster are sent using the global MAC address, so they can be received by any of the units at both sites; filters at the OTV localize the traffic within the data center.



Spanned EtherChannel Transparent Mode North-South Inter-Site Example

The following example shows 2 cluster members at each of 2 data centers placed between inside and outside routers (North-South insertion). The cluster members are connected by the cluster control link over the DCI. The cluster members at each site connect to the local switches using spanned EtherChannels for the inside and outside. Each EtherChannel is spanned across all chassis in the cluster.

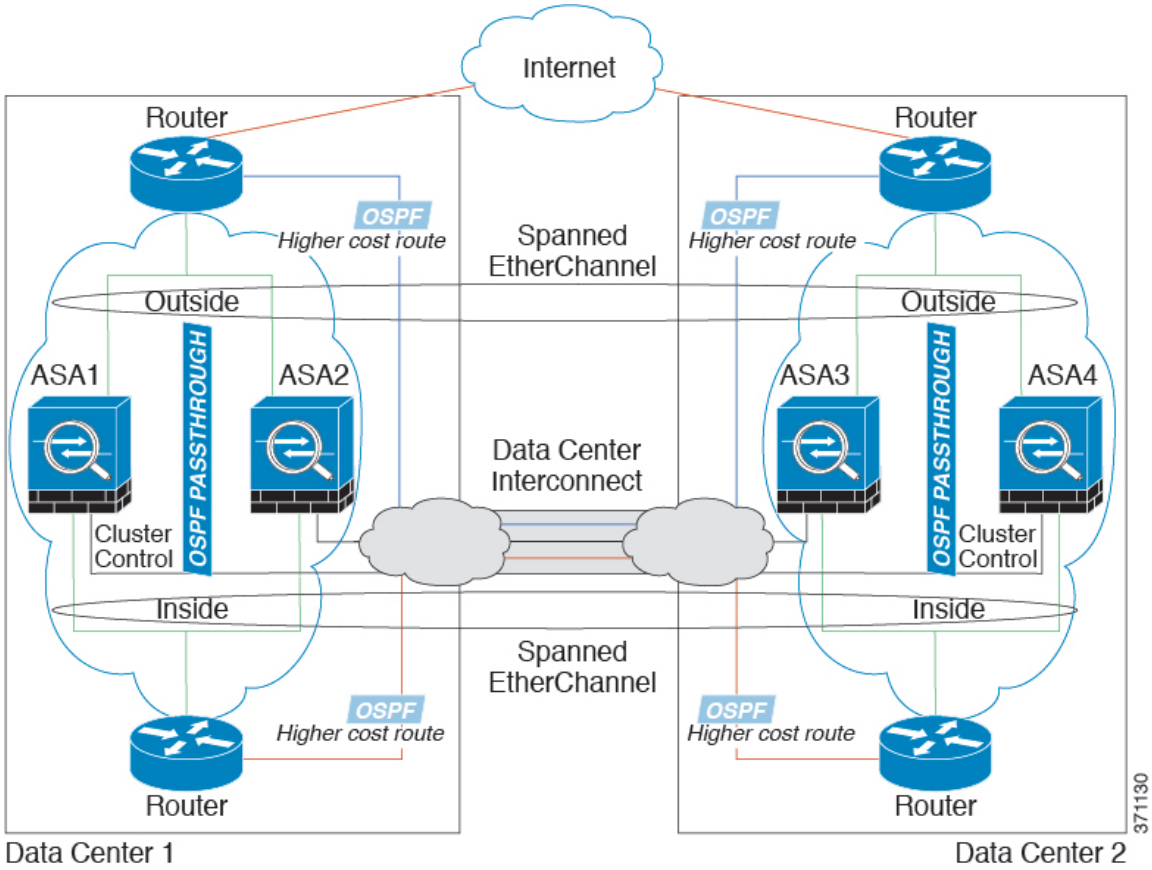
The inside and outside routers at each data center use OSPF, which is passed through the transparent ASAs. Unlike MACs, router IPs are unique on all routers. By assigning a higher cost route across the DCI, traffic stays within each data center unless all cluster members at a given site go down. The lower cost route through the ASAs must traverse the same bridge group at each site for the cluster to maintain asymmetric connections. In the event of a failure of all cluster members at one site, traffic goes from each router over the DCI to the cluster members at the other site.

The implementation of the switches at each site can include:

- Inter-site VSS/vPC—In this scenario, you install one switch at Data Center 1, and the other at Data Center 2. One option is for the cluster units at each Data Center to only connect to the local switch, while the VSS/vPC traffic goes across the DCI. In this case, connections are for the most part kept local to each

datacenter. You can optionally connect each unit to both switches across the DCI if the DCI can handle the extra traffic. In this case, traffic is distributed across the data centers, so it is essential for the DCI to be very robust.

- Local VSS/vPC at each site—For better switch redundancy, you can install 2 separate VSS/vPC pairs at each site. In this case, although the cluster units still have a spanned EtherChannel with Data Center 1 chassis connected only to both local switches, and Data Center 2 chassis connected to those local switches, the spanned EtherChannel is essentially "split." Each local VSS/vPC sees the spanned EtherChannel as a site-local EtherChannel.

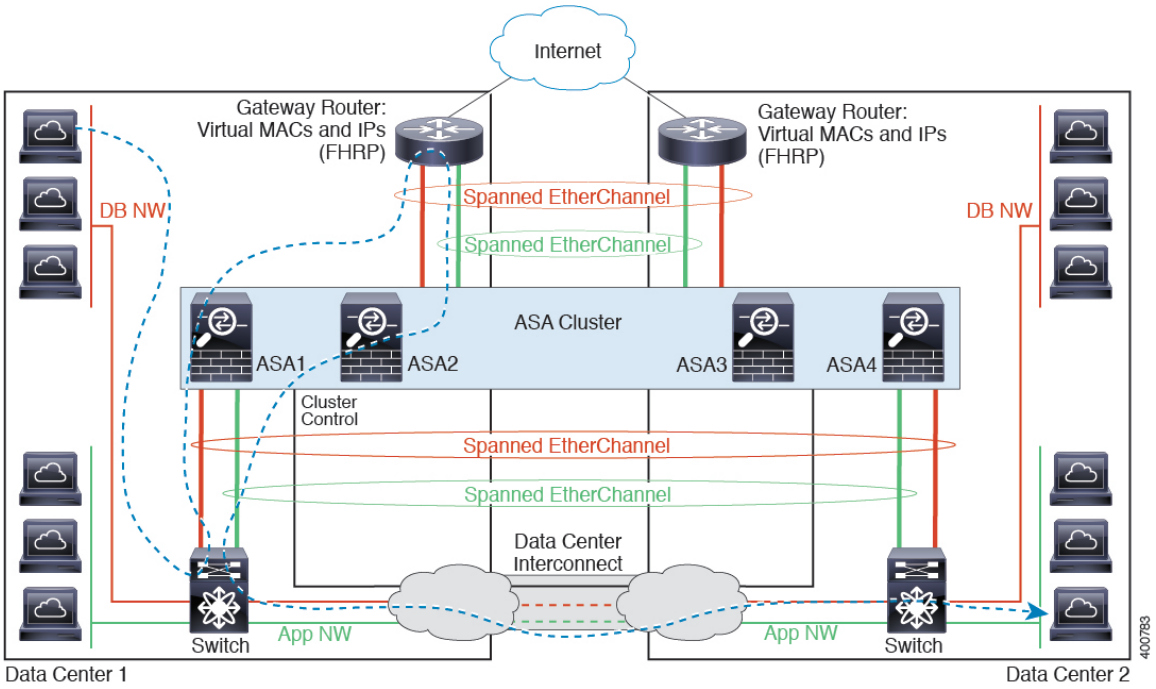


Spanned EtherChannel Transparent Mode East-West Inter-Site Example

The following example shows 2 cluster members at each of 2 data centers placed between the gateway router and two inside networks at each site, the App network and the DB network (East-West insertion). The cluster members are connected by the cluster control link over the DCI. The cluster members at each site connect to the local switches using spanned EtherChannels for both the App and DB networks on the inside and outside. Each EtherChannel is spanned across all chassis in the cluster.

The gateway router at each site uses an FHRP such as HSRP to provide the same destination virtual MAC and IP addresses at each site. A good practice to avoid unintended MAC address flapping is to statically add the gateway routers real MAC addresses to the ASA MAC address table using the **mac-address-table static** *outside_interface mac_address* command. Without these entries, if the gateway at site 1 communicates with the gateway at site 2, that traffic might pass through the ASA and attempt to reach site 2 from the inside

interface and cause problems. The data VLANs are extended between the sites using Overlay Transport Virtualization (OTV) (or something similar). You must add filters to prevent traffic from traversing the DCI to the other site when the traffic is destined for the gateway router. If the gateway router at one site becomes unreachable, you must remove the filters so traffic can be sent to the other site's gateway router.

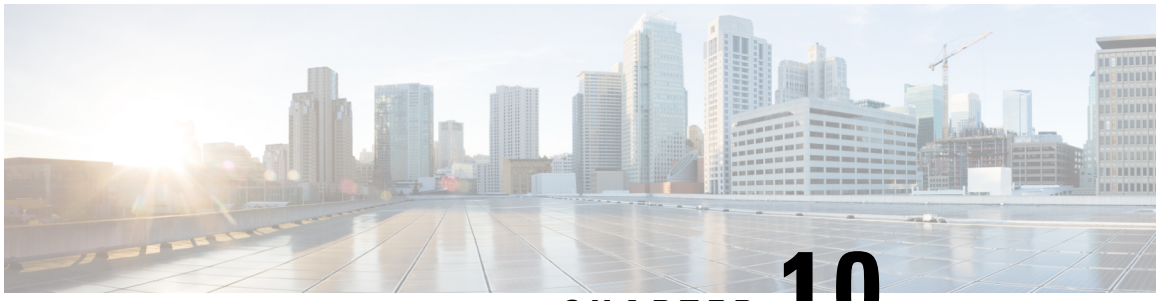


See Spanned EtherChannel Transparent Mode North-South Inter-Site Example, on page 178 for information about vPC/VSS options.

History for Logical Devices

Feature Name	Platform Releases	Feature Information
Support for FTD clustering on the Firepower 4100	2.1.1	You can cluster up to 6 chassis in an FTD cluster.
Support for ASA clustering on the Firepower 4100	1.1.4	You can cluster up to 6 chassis in an ASA cluster.
Support for intra-chassis clustering on the FTD on the Firepower 9300	1.1.4	The Firepower 9300 supports intra-chassis clustering with the FTD application. We introduced the following commands: **enter mgmt-bootstrap ftd, enter bootstrap-key FIREPOWER_MANAGER_IP, enter bootstrap-key FIREWALL_MODE, enter bootstrap-key-secret REGISTRATION_KEY, enter bootstrap-key-secret PASSWORD, enter bootstrap-key FQDN, enter bootstrap-key DNS_SERVERS, enter bootstrap-key SEARCH_DOMAINS, enter ipv4 firepower, enter ipv6 firepower, set value, set gateway, set ip, accept-license-agreement**

Feature Name	Platform Releases	Feature Information
Inter-chassis clustering for 16 ASA modules on the Firepower 9300	1.1.3	You can now enable inter-chassis clustering for the ASA. You can include up to 16 modules. For example, you can use 1 module in 16 chassis, or 2 modules in 8 chassis, or any combination that provides a maximum of 16 modules.
Intra-chassis Clustering for the ASA on the Firepower 9300	1.1.1	You can cluster all ASA security modules within the Firepower 9300 chassis. We introduced the following commands: **enter cluster-bootstrap, enter logical-device clustered, set chassis-id, set ipv4 gateway, set ipv4 pool, set ipv6 gateway, set ipv6 pool, set key, set mode spanned-etherchannel, set port-type cluster, set service-type, set virtual ipv4, set virtual ipv6**



CHAPTER 10

Configuration Import/Export

About Configuration Import/Export

You can use the configuration export feature to export an XML file containing logical device and platform configuration settings for your Firepower 4100/9300 chassis to a remote server. You can later import that configuration file to quickly apply the configuration settings to your Firepower 4100/9300 chassis to return to a known good configuration or to recover from a system failure.

Guidelines and Restrictions

- Do not modify the contents of the configuration file. If a configuration file is modified, configuration import using that file might fail.
- Application-specific configuration settings are not contained in the configuration file. You must use the configuration backup tools provided by the application to manage application-specific settings and configurations.
- When you import a configuration to the Firepower 4100/9300 chassis, all existing configuration on the Firepower 4100/9300 chassis (including any logical devices) are deleted and completely replaced by the configuration contained in the import file.
- Except in an RMA scenario, we recommend you only import a configuration file to the same Firepower 4100/9300 chassis where the configuration was exported.
- The platform software version of the Firepower 4100/9300 chassis where you are importing should be the same version as when the export was taken. If not, the import operation is not guaranteed to be successful. We recommend you export a backup configuration whenever the Firepower 4100/9300 chassis is upgraded or downgraded.
- The Firepower 4100/9300 chassis where you are importing must have the same Network Modules installed in the same slots as when the export was taken.

- The Firepower 4100/9300 chassis where you are importing must have the correct software application images installed for any logical devices defined in the export file that you are importing.
- If the configuration file being imported contains a logical device whose application has an End-User License Agreement (EULA), you must accept the EULA for that application on the Firepower 4100/9300 chassis before you import the configuration or the operation will fail.
- To avoid overwriting existing backup files, change the file name in the backup operation or copy the existing file to another location.

Exporting an FXOS Configuration File

Use the configuration export feature to export an XML file containing logical device and platform configuration settings for your Firepower 4100/9300 chassis to a remote server.

Before you begin

Review the About Configuration Import/Export.

Procedure

Step 1 To export a configuration file to a remote server:

scope system

export-config *URL* **enabled**
commit-buffer

Specify the URL for the file being exported using one of the following syntax:

- **ftp**://*username@hostname*/*path*/*image_name*
- **scp**://*username@hostname*/*path*/*image_name*
- **sftp**://*username@hostname*/*path*/*image_name*
- **tftp**://*hostname***:***port-num*/*path*/*image_name*

Note You must specify the full path including filename. If you do not specify a filename, a hidden file is created in the specified path.

Example:

```
Firepower-chassis# scope system
Firepower-chassis /system # export-config scp://user1@192.168.1.2:/export/cfg-backup.xml
enabled
Firepower-chassis /system/export-config # commit-buffer
```

Step 2 To check the status of the export task:

scope system

scope export-config *hostname*

show fsm status

Example:

```
Firepower-chassis# scope system
Firepower-chassis /system # scope export-config 192.168.1.2
Firepower-chassis /system/export-config # show fsm status

Hostname: 192.168.1.2

    FSM 1:
        Remote Result: Not Applicable
        Remote Error Code: None
        Remote Error Description:
        Status: Nop
        Previous Status: Backup Success
        Timestamp: 2016-01-03T15:32:08.636
        Try: 0
        Progress (%): 100
        Current Task:
```

Step 3 To view existing export tasks:

scope system

show export-config

Step 4 To modify an existing export task:

scope system

scope export-config *hostname*

Use the following commands to modify the export task:

- {**enable**|**disable**}
- **set description** <*description*>
- **set password** <*password*>
- **set port** <*port*>
- **set protocol** {**ftp**|**scp**|**sftp**|**tftp**}
- **set remote-file** *path_and_filename*
- **set user** <*user*>

Step 5 To delete an export task:

scope system

delete export-config *hostname*

commit-buffer

Scheduling Automatic Configuration Export

Use the scheduled export feature to automatically export an XML file containing logical device and platform configuration settings for your Firepower 4100/9300 chassis to a remote server. You can schedule the exports to be run daily, weekly, or every two weeks. The configuration export will be executed according to the schedule based on the when the scheduled export feature is enabled. So, for example, if you enable weekly scheduled export on a Wednesday at 10:00pm, the system will trigger a new export every Wednesday at 10:00pm.

Please review the About Configuration Import/Export for important information about using the configuration export feature.

Procedure

To create a scheduled export task:

a) Set the scope to export policy configuration:

scope org

scope cfg-export-policy default

b) Enable the export policy:

set adminstate enable

c) Specify the protocol to use when communicating with the remote server:

set protocol {**ftp**|**scp**|**sftp**|**tftp**}

d) Specify the hostname or IP address of the location where the backup file should be stored. This can be a server, storage array, local drive, or any read/write media that the Firepower 4100/9300 chassis can access through the network.

If you use a hostname rather than an IP address, you must configure a DNS server.

set hostname *hostname*

e) If you are using a non-default port, specify the port number:

set port *port*

f) Specify the username the system should use to log in to the remote server. This field does not apply if the protocol is TFTP:

set user *username*

g) Specify the password for the remote server username. This field does not apply if the protocol is TFTP:

set password *password*

h) Specify the full path to where you want the configuration file exported including the filename. If you omit the filename, the export procedure assigns a name to the file:

set remote-file *path_and_filename*

i) Specify the schedule on which you would like to have the configuration automatically exported. This can be one of the following: Daily, Weekly, or BiWeekly:

set schedule {**daily**|**weekly**|**bi-weekly**}

j) Commit the transaction to the system configuration:

commit-buffer

Example:

```
Firepower-chassis# scope org
Firepower-chassis /org # scope cfg-export-policy default
Firepower-chassis /org/cfg-export-policy # set adminstate enable
Firepower-chassis /org/cfg-export-policy* # set protocol scp
Firepower-chassis /org/cfg-export-policy* # set hostname 192.168.1.2
Firepower-chassis /org/cfg-export-policy* # set remote-file /export/cfg-backup.xml
Firepower-chassis /org/cfg-export-policy* # set user user1
Firepower-chassis /org/cfg-export-policy* # set password
Password:
Firepower-chassis /org/cfg-export-policy* # set schedule weekly
Firepower-chassis /org/cfg-export-policy* # commit-buffer
Firepower-chassis /org/cfg-export-policy #
Firepower-chassis /org/cfg-export-policy # show detail

Config Export policy:
    Name: default
    Description: Configuration Export Policy
    Admin State: Enable
    Protocol: Scp
    Hostname: 192.168.1.2
    User: user1
    Remote File: /export/cfg-backup.xml
    Schedule: Weekly
    Port: Default
    Current Task:
```

Setting a Configuration Export Reminder

Use the Export Reminder feature to have the system generate a fault when a configuration export hasn't been executed in a certain number of days.

Procedure

To create a configuration export reminder:

scope org

scope cfg-export-reminder

set frequency *days*

set adminstate {**enable**|**disable**}

commit-buffer

Example:

```
Firepower-chassis# scope org
Firepower-chassis /org # scope cfg-export-reminder
Firepower-chassis /org/cfg-export-reminder # set frequency 10
Firepower-chassis /org/cfg-export-reminder* # set adminstate enable
Firepower-chassis /org/cfg-export-reminder* # commit-buffer
Firepower-chassis /org/cfg-export-reminder # show detail

Config Export Reminder:
    Config Export Reminder (Days): 10
    AdminState: Enable
```

Importing a Configuration File

You can use the configuration import feature to apply configuration settings that were previously exported from your Firepower 4100/9300 chassis. This feature allows you to return to a known good configuration or to recover from a system failure.

Before you begin

Review the About Configuration Import/Export.

Procedure

Step 1 To import a configuration file from a remote server:

scope system

import-config *URL* **enabled**

commit-buffer

Specify the URL for the file being imported using one of the following syntax:

- **ftp**://*username@hostname*/*path*/*image_name*
- **scp**://*username@hostname*/*path*/*image_name*
- **sftp**://*username@hostname*/*path*/*image_name*
- **tftp**://*hostname***:***port-num*/*path*/*image_name*

Example:

```
Firepower-chassis# scope system
Firepower-chassis /system # import-config scp://user1@192.168.1.2:/import/cfg-backup.xml
enabled
Warning: After configuration import any changes on the breakout port configuration will
  cause the system to reboot
Firepower-chassis /system/import-config # commit-buffer
```

Step 2 To check the status of the import task:

scope system

scope import-config *hostname*

show fsm status

Example:

```
Firepower-chassis# scope system
Firepower-chassis /system # scope import-config 192.168.1.2
Firepower-chassis /system/import-config # show fsm status

Hostname: 192.168.1.2

    FSM 1:
        Remote Result: Not Applicable
        Remote Error Code: None
        Remote Error Description:
        Status: Import Wait For Switch
        Previous Status: Import Config Breakout
        Timestamp: 2016-01-03T15:45:03.963
        Try: 0
        Progress (%): 97
        Current Task: updating breakout port configuration(FSM-STAGE:sam:dme:
          MgmtImporterImport:configBreakout)
```

Step 3 To view existing import tasks:

scope system

show import-config

Step 4 To modify an existing import task:

scope system

scope import-config *hostname*

Use the following commands to modify the import task:

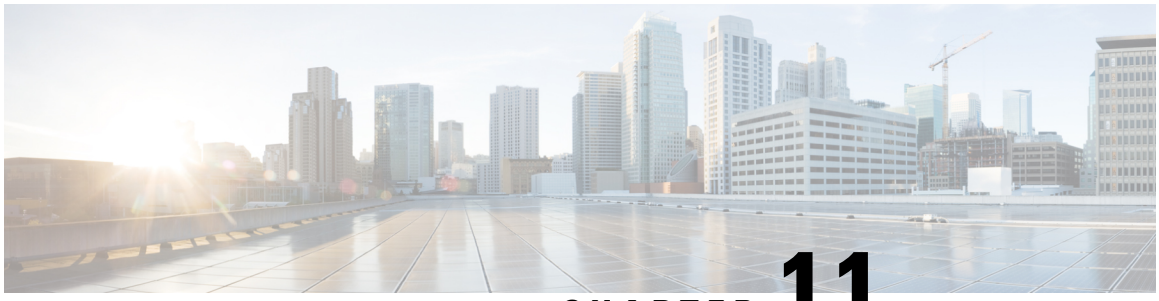
- {**enable**|**disable**}
- **set description** <*description*>
- **set password** <*password*>
- **set port** <*port*>
- **set protocol** {**ftp**|**scp**|**sftp**|**tftp**}
- **set remote-file** *path_and_filename*
- **set user** <*user*>

Step 5 To delete an import task:

scope system

delete import-config *hostname*

commit-buffer



CHAPTER 11

Packet Capture

Packet Capture

The Packet Capture tool is a valuable asset for use in debugging connectivity and configuration issues and for understanding traffic flows through your Firepower 4100/9300 chassis. You can use the Packet Capture tool to log traffic that is going through specific interfaces on your Firepower 4100/9300 chassis.

You can create multiple packet capture sessions, and each session can capture traffic on multiple interfaces. For each interface included in a packet capture session, a separate packet capture (PCAP) file will be created.

Backplane Port Mappings

The Firepower 4100/9300 chassis uses the following mappings for internal backplane ports:

Security Module	Port Mapping	Description
Security Module 1/Security Engine	Ethernet1/9	Internal-Data0/0
Security Module 1/Security Engine	Ethernet1/10	Internal-Data0/1
Security Module 2	Ethernet1/11	Internal-Data0/0
Security Module 2	Ethernet1/12	Internal-Data0/1
Security Module 3	Ethernet1/13	Internal-Data0/0
Security Module 3	Ethernet1/14	Internal-Data0/1

Guidelines and Limitations for Packet Capture

The Packet Capture tool has the following limitations:

- Can capture only up to 100 Mbps.
- Packet capture sessions can be created even when there is not enough storage space available to run the packet capture session. You should verify that you have enough storage space available before you start a packet capture session.
- Does not support multiple active packet capturing sessions.
- There is no option to filter based on source or destination IPv6 address.
- Captures only at the ingress stage of the internal switch.
- Filters are not effective on packets that cannot be understood by the internal switch (for example Security Group Tag and Network Service Header packets).
- You cannot capture packets for an EtherChannel as a whole. However, for an EtherChannel allocated to a logical device, you can capture packets on each member interface of the EtherChannel.
- You cannot copy or export a PCAP file while the capture session is still active.
- When you delete a packet capture session, all packet capture files associated with that session are also deleted.

Creating or Editing a Packet Capture Session

Procedure

Step 1 Enter packet capture mode:

Firepower-chassis # **scope packet-capture**

Step 2 Create a filter; see Configuring Filters for Packet Capture, on page 194.

You can apply filters to any of the interfaces included in a packet capture session.

Step 3 To create or edit a packet capture session:

Firepower-chassis /packet-capture # **enter session** *session_name*

Step 4 Specify the buffer size to use for this packet capture session:

Firepower-chassis /packet-capture/session* # **set session-memory-usage** *session_size_in_megabytes*

The specified buffer size must be between 256 and 2048 MB.

Step 5 Specify the physical source ports that should be included in this packet capture session.

You can capture from multiple ports and can capture from both physical ports and application ports during the same packet capture session. A separate packet capture file is created for each port included in the session.

You cannot capture packets for an EtherChannel as a whole. However, for an EtherChannel allocated to a logical device, you can capture packets on each member interface of the EtherChannel.

Note To remove a port from the packet capture session, use **delete** instead of **create** in the commands listed below.

a) Specify the physical port.

Firepower-chassis /packet-capture/session* # **create** {**phy-port | phy-aggr-port**} *port_id*

Example:

```
Firepower-chassis /packet-capture/session* # create phy-port Ethernet1/1
Firepower-chassis /packet-capture/session/phy-port* #
```

b) (Optional) Apply the desired filter.

Firepower-chassis /packet-capture/session/phy-port* # **set** {**source-filter**} *filtername*

Note To remove a filter from a port, use **set source-filter ""**.

c) Repeat the steps above as needed to add all desired ports.

Step 6 Specify the application source ports that should be included in this packet capture session.

You can capture from multiple ports and can capture from both physical ports and application ports during the same packet capture session. A separate packet capture file is created for each port included in the session.

Note To remove a port from the packet capture session, use **delete** instead of **create** in the commands listed below.

a) Specify the application port.

Firepower-chassis /packet-capture/session* # **create app_port** *module_slot link_name interface_name app_name*

b) (Optional) Apply the desired filter.

Firepower-chassis /packet-capture/session/phy-port* # **set** {**source-filter**} *filtername*

Note To remove a filter from a port, use **set source-filter ""**.

c) Repeat the steps above as needed to add all desired application ports.

Step 7 If you want to start the packet capture session now:

Firepower-chassis /packet-capture/session* # **enable**

Newly created packet-capture sessions are disabled by default. Explicit enabling of a session activates the packet capture session when the changes are committed. If another session is already active, enabling a session will generate an error. You must disable the already active packet-capture session before you can enable this session.

Step 8 Commit the transaction to the system configuration:

Firepower-chassis /packet-capture/session* # **commit-buffer**

If you enabled the packet capture session, the system will begin capturing packets. You will need to stop capturing before you can download the PCAP files from your session.

Example

```
Firepower-chassis# scope packet-capture
Firepower-chassis packet-capture # create session asa1inside
Firepower-chassis packet-capture/session # set session-memory-usage 256
Firepower-chassis packet-capture/session* # create phy-port Ethernet3/1
Firepower-chassis packet-capture/session* # create phy-aggr-port Ethernet2/1/1
Firepower-chassis packet-capture/session* # create app-port 1 link1 Ethernet 1/1 asa
Firepower-chassis packet-capture/session* # create filter interface1vlan100
Firepower-chassis packet-capture/filter* # set ivlan 100
Firepower-chassis packet-capture/filter* # set srcIP 6.6.6.6
Firepower-chassis packet-capture/filter* # set srcPort 80
Firepower-chassis packet-capture/filter* # set destIP 10.10.10.10
Firepower-chassis packet-capture/filter* # set destPort 5050
Firepower-chassis packet-capture/filter* # exit
Firepower-chassis packet-capture/session* # scope phy-port Ethernet3/1
Firepower-chassis packet-capture/session/phy-port* # set src-filter interface1vlan100
Firepower-chassis packet-capture/session/phy-port* # exit
Firepower-chassis packet-capture/session* # scope app-port 1 link1 Ethernet1/1 asa
Firepower-chassis packet-capture/session/app-port* # set src-filter interface1vlan100
Firepower-chassis packet-capture/session/app-port* # exit
Firepower-chassis packet-capture/session* # enable
Firepower-chassis packet-capture/session* # commit-buffer
Firepower-chassis packet-capture/session #
```

Configuring Filters for Packet Capture

You can create filters to limit the traffic that is included in a packet capture session. You can select which interfaces should use a specific filter while creating a packet capture session.

Note If you modify or delete a filter that is applied to a packet capture session that is currently running, the changes will not take affect until you disable that session and then reenable it.

Procedure

Step 1 Enter packet capture mode:

Firepower-chassis # **scope packet-capture**

Step 2 To create a new packet capture filter:

Firepower-chassis /packet-capture # **create filter** *filter_name*

To edit an existing packet capture filter:

Firepower-chassis /packet-capture # **enter filter** *filter_name*

To delete an existing packet capture filter:

Firepower-chassis /packet-capture # **delete filter** *filter_name*

Step 3 Specify the filter details by setting one or more filter properties:

Firepower-chassis /packet-capture/filter* # **set** <*filterprop filterprop_value*

Table 5: Supported Filter Properties

ivlan	Inner VLAN ID (vlan of packet while ingressing port)
ovlan	Outer VLAN ID (vlan added by the Firepower 4100/9300 chassis)
srcip	Source IP Address (IPv4)
destip	Destination IP Address (IPv4)
srcport	Source Port Number
destport	Destination Port Number
protocol	IP Protocol [IANA defined Protocol values in decimal format]
ethertype	Ethernet Protocol type [IANA defined Ethernet Protocol type value in decimal format. For eg: IPv4 = 2048, IPv6 = 34525, ARP = 2054, SGT = 35081]
srcmac	Source Mac Address
destmac	Destination Mac Address

Example

```
Firepower-chassis# scope packet-capture
Firepower-chassis packet-capture # create filter interface1vlan100
Firepower-chassis packet-capture/filter* # set ivlan 100
Firepower-chassis packet-capture/filter* # set srcip 6.6.6.6
Firepower-chassis packet-capture/filter* # set srcport 80
Firepower-chassis packet-capture/filter* # set destip 10.10.10.10
Firepower-chassis packet-capture/filter* # set destport 5050
Firepower-chassis packet-capture/filter* # commit-buffer
```

Starting and Stopping a Packet Capture Session

Procedure

Step 1 Enter packet capture mode:

Firepower-chassis # **scope packet-capture**

Step 2 Enter the scope for the packet capture session that you want to start or stop:

Firepower-chassis /packet-capture # **enter session** *session_name*

Step 3 To start a packet capture session:

Firepower-chassis /packet-capture/session* # **enable** [append | overwrite]

Note You cannot start a packet capture session while another session is running.

While the packet capture session is running, the file size for the individual PCAP files will increase as traffic is captured. Once the Buffer Size limit is reached, the system will start dropping packets and you will see the Drop Count field increase.

Step 4 To stop a packet capture session:

Firepower-chassis /packet-capture/session* # **disable**

Step 5 Commit the transaction to the system configuration:

Firepower-chassis /packet-capture/session* # **commit-buffer**

If you enabled the packet capture session, the PCAP files for the interfaces included in the session will start collecting traffic. If the session is configured to overwrite session data, the existing PCAP data will be erased. If not, data will be appended to the existing file (if any).

Example

```
Firepower-chassis# scope packet-capture
Firepower-chassis packet-capture # scope session asa1inside
Firepower-chassis packet-capture/session # enable append
Firepower-chassis packet-capture/session* # commit-buffer
Firepower-chassis packet-capture/session #
```

Downloading a Packet Capture File

You can download the Packet Capture (PCAP) files from a session to your local computer so that they can be analyzed using a network packet analyzer.

PCAP files are stored into the `workspace://packet-capture` directory and use the following naming conventions:

`workspace://packet-capture/session-<id>/<session-name>-<interface-name>.pcap`

Procedure

To copy a PCAP file from the Firepower 4100/9300 chassis:

Note You should stop the packet capture session before you download the PCAP files from that session.

a) Connect to local management:

Firepower-chassis # **connect localmgmt**

b) Copy the PCAP files:

copy *pcap_file copy_destination*

Example

```
Firepower-chassis# connect localmgmt
# copy workspace:/packet-capture/session-1/test-ethernet-1-1-0.pcap
scp://user@10.10.10.1:/workspace/
```

INDEX

E

F

H

I

K

L

M

N

O

P

R

S

T

U