Using Multi-Instance Capability on the Firepower 4100/9300

Multi-instance capability lets you run container instances that use a subset of resources of the security module/engine. Multi-instance capability is only supported for the Firepower Threat Defense; it is not supported for the ASA.

For multi-instance clustering, see Deploying a Cluster for Firepower Threat Defense for Scalability and High Availability or the FXOS configuration guide.

This document covers the latest FXOS version features; see History for Multi-Instance Capability, on page 43 for details about feature changes. If you have an old version installed, see the procedures in the FXOS configuration guide for your version.

About Multi-Instance Capability

The Firepower chassis includes a supervisor and up to three security modules on which you can install logical devices. A logical device lets you run one application instance (Firepower Threat Defense or ASA). 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. The application type determines whether you can run a single instance (native) or multiple instances (container).

Logical Device Application Instances: Container and Native

Application instances run in the following deployment types:

- Native instance—A native instance uses all of the resources (CPU, RAM, and disk space) of the security module/engine, so you can only install one native instance.

- Container instance—A container instance uses a subset of resources of the security module/engine, so you can install multiple container instances. Multi-instance capability is only supported for the Firepower Threat Defense using FMC; it is not supported for the ASA or the FTD using FDM.
Multi-instance capability is similar to ASA multiple context mode, although the implementation is different. Multiple context mode partitions a single application instance, while multi-instance capability allows independent container instances. Container instances allow hard resource separation, separate configuration management, separate reloads, separate software updates, and full Firepower Threat Defense feature support. Multiple context mode, due to shared resources, supports more contexts on a given platform. Multiple context mode is not available on the Firepower Threat Defense.

For the Firepower 9300, you can use a native instance on some modules, and container instances on the other module(s).

**Container Instance Interfaces**

To provide flexible physical interface use for container instances, you can create VLAN subinterfaces in FXOS and also share interfaces (VLAN or physical) between multiple instances. Native instances cannot use VLAN subinterfaces or shared interfaces. A multi-instance cluster cannot use VLAN subinterfaces or shared interfaces. An exception is made for the cluster control link, which can use a subinterface of the Cluster EtherChannel. See [Shared Interface Scalability, on page 3](#) and [Add a VLAN Subinterface for Container Instances, on page 30](#).

**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.

- **Data-sharing**—Use for regular data. Only supported with container instances, these data interfaces can be shared by one or more logical devices/container instances (FTD-using-FMC only). Each container instance can communicate over the backplane with all other instances that share this interface. Shared interfaces can affect the number of container instances you can deploy. Shared interfaces are not supported for bridge group member interfaces (in transparent mode or routed mode), inline sets, passive interfaces, clusters, or failover links.

- **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, FMC in 6.7 and later, and FDM: 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.

- **Firepower-eventing**—Use as a secondary management interface for FTD-using-FMC 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. If you later configure a data interface for management, you cannot use a separate eventing 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. For multi-instance clustering, you cannot share a Cluster-type interface across devices. You can add VLAN subinterfaces to the Cluster EtherChannel to provide separate cluster control links per cluster. If you add subinterfaces to a Cluster interface, you cannot use that interface for a native cluster. FDM does not support clustering.

See the following table for interface type support for FTD and ASA applications in standalone and cluster deployments.

**Table 1: Interface Type Support**

<table>
<thead>
<tr>
<th>Application</th>
<th>Data: Subinterface</th>
<th>Data-Sharing Subinterface</th>
<th>Mgmt</th>
<th>FirepowerMgmt</th>
<th>Cluster: (EtherChannel only)</th>
<th>Cluster: Subinterface</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FTD</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Standalone Native Instance</td>
<td>Yes</td>
<td>—</td>
<td>—</td>
<td>Yes</td>
<td>Yes</td>
<td>—</td>
</tr>
<tr>
<td>Standalone Container Instance</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>—</td>
</tr>
<tr>
<td>Cluster Native Instance (EtherChannel only for inter-chassis cluster)</td>
<td>Yes</td>
<td>—</td>
<td>—</td>
<td>Yes</td>
<td>Yes</td>
<td>—</td>
</tr>
<tr>
<td>Cluster Container Instance (EtherChannel only for inter-chassis cluster)</td>
<td>Yes</td>
<td>—</td>
<td>—</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>ASA</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Standalone Native Instance</td>
<td>Yes</td>
<td>—</td>
<td>—</td>
<td>Yes</td>
<td>—</td>
<td>Yes</td>
</tr>
<tr>
<td>Cluster Native Instance (EtherChannel only for inter-chassis cluster)</td>
<td>Yes</td>
<td>—</td>
<td>—</td>
<td>Yes</td>
<td>—</td>
<td>Yes</td>
</tr>
</tbody>
</table>

**Shared Interface Scalability**

Container instances can share data-sharing type interfaces. This capability lets you conserve physical interface usage as well as support flexible networking deployments. When you share an interface, the chassis uses
unique MAC addresses to forward traffic to the correct instance. However, shared interfaces can cause the forwarding table to grow large due to the need for a full mesh topology within the chassis (every instance must be able to communicate with every other instance that is sharing the same interface). Therefore, there are limits to how many interfaces you can share.

In addition to the forwarding table, the chassis maintains a VLAN group table for VLAN subinterface forwarding. You can create up to 500 VLAN subinterfaces.

See the following limits for shared interface allocation:

**Max. 10 shared interfaces per instance**

**Max. 14 instances per shared interface**

---

**Shared Interface Best Practices**

For optimal scalability of the forwarding table, share as few interfaces as possible. Instead, you can create up to 500 VLAN subinterfaces on one or more physical interfaces, and then divide the VLANs among the container instances.

When sharing interfaces, follow these practices in the order of most scalable to least scalable:

1. **Best**—Share subinterfaces under a single parent, and use the same set of subinterfaces with the same group of instances.

For example, create a large EtherChannel to bundle all of your like-kind interfaces together, and then share subinterfaces of that EtherChannel: Port-Channel1.2, 3, and 4 instead of Port-Channel2, Port-Channel3, and Port-Channel4. When you share subinterfaces from a single parent, the VLAN group table provides better scaling of the forwarding table than when sharing physical/EtherChannel interfaces or subinterfaces across parents.
If you do not share the same set of subinterfaces with a group of instances, your configuration can cause more resource usage (more VLAN groups). For example, share Port-Channel1.2, 3, and 4 with instances 1, 2, and 3 (one VLAN group) instead of sharing Port-Channel1.2 and 3 with instances 1 and 2, while sharing Port-Channel1.3 and 4 with instance 3 (two VLAN groups).

2. Fair—Share subinterfaces across parents.

For example, share Port-Channel1.2, Port-Channel2.3, and Port-Channel3.4 instead of Port-Channel2, Port-Channel4, and Port-Channel4. Although this usage is not as efficient as only sharing subinterfaces on the same parent, it still takes advantage of VLAN groups.
3. Worst—Share individual parent interfaces (physical or EtherChannel).
This method uses the most forwarding table entries.

**Figure 3: Fair: Shared Subinterfaces on Separate Parents**

<table>
<thead>
<tr>
<th>Port-channel1</th>
<th>Port-channel2</th>
<th>Port-channel3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data</td>
<td>Data</td>
<td>Data</td>
</tr>
<tr>
<td>VLAN 2</td>
<td>VLAN 3</td>
<td>VLAN 4</td>
</tr>
</tbody>
</table>

**Figure 4: Worst: Shared Parent Interfaces**

<table>
<thead>
<tr>
<th>Ethernet1/1</th>
<th>Ethernet1/2</th>
<th>Ethernet1/3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data-sharing</td>
<td>Data-sharing</td>
<td>Data-sharing</td>
</tr>
</tbody>
</table>

**Shared Interface Usage Examples**

See the following tables for examples of interface sharing and scalability. The below scenarios assume use of one physical/EtherChannel interface for management shared across all instances, and another physical or EtherChannel interface with dedicated subinterfaces for use with High Availability.

- Table 2: Physical/EtherChannel Interfaces and Instances on a Firepower 9300 with Three SM-44s, on page 7
- Table 3: Subinterfaces on One Parent and Instances on a Firepower 9300 with Three SM-44s, on page 8
- Table 4: Physical/EtherChannel Interfaces and Instances on a Firepower 9300 with One SM-44, on page 10
- Table 5: Subinterfaces on One Parent and Instances on a Firepower 9300 with One SM-44, on page 11

**Firepower 9300 with Three SM-44s**

The following table applies to three SM-44 security modules on a 9300 using only physical interfaces or EtherChannels. Without subinterfaces, the maximum number of interfaces are limited. Moreover, sharing multiple physical interfaces uses more forwarding table resources than sharing multiple subinterfaces.
Each SM-44 module can support up to 14 instances. Instances are split between modules as necessary to stay within limits.

Table 2: Physical/EtherChannel Interfaces and Instances on a Firepower 9300 with Three SM-44s

<table>
<thead>
<tr>
<th>Dedicated Interfaces</th>
<th>Shared Interfaces</th>
<th>Number of Instances</th>
<th>% Forwarding Table Used</th>
</tr>
</thead>
<tbody>
<tr>
<td>32:</td>
<td>0</td>
<td>4:</td>
<td>16%</td>
</tr>
<tr>
<td>• 8</td>
<td>• Instance 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• 8</td>
<td>• Instance 2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• 8</td>
<td>• Instance 3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• 8</td>
<td>• Instance 4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>30:</td>
<td>0</td>
<td>2:</td>
<td>14%</td>
</tr>
<tr>
<td>• 15</td>
<td>• Instance 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• 15</td>
<td>• Instance 2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>14:</td>
<td>1</td>
<td>14:</td>
<td>46%</td>
</tr>
<tr>
<td>• 14 (1 ea.)</td>
<td>• Instance 1-Instance 14</td>
<td></td>
<td></td>
</tr>
<tr>
<td>33:</td>
<td>3:</td>
<td>33:</td>
<td>98%</td>
</tr>
<tr>
<td>• 11 (1 ea.)</td>
<td>• 1</td>
<td>• Instance 1-Instance 11</td>
<td></td>
</tr>
<tr>
<td>• 11 (1 ea.)</td>
<td>• 1</td>
<td>• Instance 12-Instance 22</td>
<td></td>
</tr>
<tr>
<td>• 11 (1 ea.)</td>
<td>• 1</td>
<td>• Instance 23-Instance 33</td>
<td></td>
</tr>
<tr>
<td>33:</td>
<td>3:</td>
<td>34:</td>
<td>DISALLOWED</td>
</tr>
<tr>
<td>• 11 (1 ea.)</td>
<td>• 1</td>
<td>• Instance 1-Instance 11</td>
<td></td>
</tr>
<tr>
<td>• 11 (1 ea.)</td>
<td>• 1</td>
<td>• Instance 12-Instance 22</td>
<td></td>
</tr>
<tr>
<td>• 12 (1 ea.)</td>
<td>• 1</td>
<td>• Instance 23-Instance 34</td>
<td></td>
</tr>
<tr>
<td>30:</td>
<td>1</td>
<td>6:</td>
<td>25%</td>
</tr>
<tr>
<td>• 30 (1 ea.)</td>
<td>• Instance 1-Instance 6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>30:</td>
<td>3:</td>
<td>6:</td>
<td>23%</td>
</tr>
<tr>
<td>• 10 (5 ea.)</td>
<td>• 1</td>
<td>• Instance 1-Instance 2</td>
<td></td>
</tr>
<tr>
<td>• 10 (5 ea.)</td>
<td>• 1</td>
<td>• Instance 2-Instance 4</td>
<td></td>
</tr>
<tr>
<td>• 10 (5 ea.)</td>
<td>• 1</td>
<td>• Instance 5-Instance 6</td>
<td></td>
</tr>
<tr>
<td>Dedicated Interfaces</td>
<td>Shared Interfaces</td>
<td>Number of Instances</td>
<td>% Forwarding Table Used</td>
</tr>
<tr>
<td>----------------------</td>
<td>------------------</td>
<td>---------------------</td>
<td>-------------------------</td>
</tr>
<tr>
<td>30:</td>
<td>2</td>
<td>5:</td>
<td>28%</td>
</tr>
<tr>
<td>• 30 (6 ea.)</td>
<td></td>
<td>• Instance 1-Instance 5</td>
<td></td>
</tr>
<tr>
<td>30:</td>
<td>4: 2</td>
<td>5:</td>
<td>26%</td>
</tr>
<tr>
<td>• 12 (6 ea.)</td>
<td>• 2</td>
<td>• Instance 1-Instance 2</td>
<td></td>
</tr>
<tr>
<td>• 18 (6 ea.)</td>
<td>• 2</td>
<td>• Instance 2-Instance 5</td>
<td></td>
</tr>
<tr>
<td>24:</td>
<td>7</td>
<td>4:</td>
<td>44%</td>
</tr>
<tr>
<td>• 6</td>
<td></td>
<td>• Instance 1</td>
<td></td>
</tr>
<tr>
<td>• 6</td>
<td></td>
<td>• Instance 2</td>
<td></td>
</tr>
<tr>
<td>• 6</td>
<td></td>
<td>• Instance 3</td>
<td></td>
</tr>
<tr>
<td>• 6</td>
<td></td>
<td>• Instance 4</td>
<td></td>
</tr>
<tr>
<td>24:</td>
<td>14: 7</td>
<td>4:</td>
<td>41%</td>
</tr>
<tr>
<td>• 12 (6 ea.)</td>
<td>• 7</td>
<td>• Instance 1-Instance 2</td>
<td></td>
</tr>
<tr>
<td>• 12 (6 ea.)</td>
<td>• 7</td>
<td>• Instance 2-Instance 4</td>
<td></td>
</tr>
</tbody>
</table>

The following table applies to three SM-44 security modules on a 9300 using subinterfaces on a single parent physical interface. For example, create a large EtherChannel to bundle all of your like-kind interfaces together, and then share subinterfaces of that EtherChannel. Sharing multiple physical interfaces uses more forwarding table resources than sharing multiple subinterfaces.

Each SM-44 module can support up to 14 instances. Instances are split between modules as necessary to stay within limits.

**Table 3: Subinterfaces on One Parent and Instances on a Firepower 9300 with Three SM-44s**

<table>
<thead>
<tr>
<th>Dedicated Subinterfaces</th>
<th>Shared Subinterfaces</th>
<th>Number of Instances</th>
<th>% Forwarding Table Used</th>
</tr>
</thead>
<tbody>
<tr>
<td>168:</td>
<td>0</td>
<td>42:</td>
<td>33%</td>
</tr>
<tr>
<td>• 168 (4 ea.)</td>
<td></td>
<td>• Instance 1-Instance 42</td>
<td></td>
</tr>
<tr>
<td>224:</td>
<td>0</td>
<td>14:</td>
<td>27%</td>
</tr>
<tr>
<td>• 224 (16 ea.)</td>
<td></td>
<td>• Instance 1-Instance 14</td>
<td></td>
</tr>
<tr>
<td>14:</td>
<td>1</td>
<td>14:</td>
<td>46%</td>
</tr>
<tr>
<td>• 14 (1 ea.)</td>
<td></td>
<td>• Instance 1-Instance 14</td>
<td></td>
</tr>
</tbody>
</table>
### Shared Interface Usage Examples

<table>
<thead>
<tr>
<th>Dedicated Subinterfaces</th>
<th>Shared Subinterfaces</th>
<th>Number of Instances</th>
<th>% Forwarding Table Used</th>
</tr>
</thead>
<tbody>
<tr>
<td>33:</td>
<td>3:</td>
<td>33:</td>
<td>98%</td>
</tr>
<tr>
<td>• 11 (1 ea.)</td>
<td>• 1</td>
<td>• Instance 1-Instance 11</td>
<td></td>
</tr>
<tr>
<td>• 11 (1 ea.)</td>
<td>• 1</td>
<td>• Instance 12-Instance 22</td>
<td></td>
</tr>
<tr>
<td>• 11 (1 ea.)</td>
<td>• 1</td>
<td>• Instance 23-Instance 33</td>
<td></td>
</tr>
<tr>
<td>70:</td>
<td>1</td>
<td>14:</td>
<td>46%</td>
</tr>
<tr>
<td>• 70 (5 ea.)</td>
<td></td>
<td>• Instance 1-Instance 14</td>
<td></td>
</tr>
<tr>
<td>165:</td>
<td>3:</td>
<td>33:</td>
<td>98%</td>
</tr>
<tr>
<td>• 55 (5 ea.)</td>
<td>• 1</td>
<td>• Instance 1-Instance 11</td>
<td></td>
</tr>
<tr>
<td>• 55 (5 ea.)</td>
<td>• 1</td>
<td>• Instance 12-Instance 22</td>
<td></td>
</tr>
<tr>
<td>• 55 (5 ea.)</td>
<td>• 1</td>
<td>• Instance 23-Instance 33</td>
<td></td>
</tr>
<tr>
<td>70:</td>
<td>2</td>
<td>14:</td>
<td>46%</td>
</tr>
<tr>
<td>• 70 (5 ea.)</td>
<td></td>
<td>• Instance 1-Instance 14</td>
<td></td>
</tr>
<tr>
<td>165:</td>
<td>6:</td>
<td>33:</td>
<td>98%</td>
</tr>
<tr>
<td>• 55 (5 ea.)</td>
<td>• 2</td>
<td>• Instance 1-Instance 11</td>
<td></td>
</tr>
<tr>
<td>• 55 (5 ea.)</td>
<td>• 2</td>
<td>• Instance 12-Instance 22</td>
<td></td>
</tr>
<tr>
<td>• 55 (5 ea.)</td>
<td>• 2</td>
<td>• Instance 23-Instance 33</td>
<td></td>
</tr>
<tr>
<td>70:</td>
<td>10</td>
<td>14:</td>
<td>46%</td>
</tr>
<tr>
<td>• 70 (5 ea.)</td>
<td></td>
<td>• Instance 1-Instance 14</td>
<td></td>
</tr>
<tr>
<td>165:</td>
<td>30:</td>
<td>33:</td>
<td>102%</td>
</tr>
<tr>
<td>• 55 (5 ea.)</td>
<td>• 10</td>
<td>• Instance 1-Instance 11</td>
<td>DISALLOWED</td>
</tr>
<tr>
<td>• 55 (5 ea.)</td>
<td>• 10</td>
<td>• Instance 12-Instance 22</td>
<td></td>
</tr>
<tr>
<td>• 55 (5 ea.)</td>
<td>• 10</td>
<td>• Instance 23-Instance 33</td>
<td></td>
</tr>
</tbody>
</table>

**Firepower 9300 with One SM-44**

The following table applies to the Firepower 9300 with one SM-44 using only physical interfaces or EtherChannels. Without subinterfaces, the maximum number of interfaces are limited. Moreover, sharing multiple physical interfaces uses more forwarding table resources than sharing multiple subinterfaces.

The Firepower Firepower 9300 with one SM-44 can support up to 14 instances.
### Table 4: Physical/EtherChannel Interfaces and Instances on a Firepower 9300 with One SM-44

<table>
<thead>
<tr>
<th>Dedicated Interfaces</th>
<th>Shared Interfaces</th>
<th>Number of Instances</th>
<th>% Forwarding Table Used</th>
</tr>
</thead>
<tbody>
<tr>
<td>32:</td>
<td>0</td>
<td>4:</td>
<td>16%</td>
</tr>
<tr>
<td>• 8</td>
<td></td>
<td>• Instance 1</td>
<td></td>
</tr>
<tr>
<td>• 8</td>
<td></td>
<td>• Instance 2</td>
<td></td>
</tr>
<tr>
<td>• 8</td>
<td></td>
<td>• Instance 3</td>
<td></td>
</tr>
<tr>
<td>• 8</td>
<td></td>
<td>• Instance 4</td>
<td></td>
</tr>
<tr>
<td>30:</td>
<td>0</td>
<td>2:</td>
<td>14%</td>
</tr>
<tr>
<td>• 15</td>
<td></td>
<td>• Instance 1</td>
<td></td>
</tr>
<tr>
<td>• 15</td>
<td></td>
<td>• Instance 2</td>
<td></td>
</tr>
<tr>
<td>14:</td>
<td>1</td>
<td>14:</td>
<td>46%</td>
</tr>
<tr>
<td>• 14 (1 ea.)</td>
<td></td>
<td>• Instance 1-Instance 14</td>
<td></td>
</tr>
<tr>
<td>14:</td>
<td>2:</td>
<td>14:</td>
<td>37%</td>
</tr>
<tr>
<td>• 7 (1 ea.)</td>
<td>• 1</td>
<td>• Instance 1-Instance 7</td>
<td></td>
</tr>
<tr>
<td>• 7 (1 ea.)</td>
<td>• 1</td>
<td>• Instance 8-Instance 14</td>
<td></td>
</tr>
<tr>
<td>32:</td>
<td>1</td>
<td>4:</td>
<td>21%</td>
</tr>
<tr>
<td>• 8</td>
<td></td>
<td>• Instance 1</td>
<td></td>
</tr>
<tr>
<td>• 8</td>
<td></td>
<td>• Instance 2</td>
<td></td>
</tr>
<tr>
<td>• 8</td>
<td></td>
<td>• Instance 3</td>
<td></td>
</tr>
<tr>
<td>• 8</td>
<td></td>
<td>• Instance 4</td>
<td></td>
</tr>
<tr>
<td>32:</td>
<td>2</td>
<td>4:</td>
<td>20%</td>
</tr>
<tr>
<td>• 16 (8 ea.)</td>
<td></td>
<td>• Instance 1-Instance 2</td>
<td></td>
</tr>
<tr>
<td>• 16 (8 ea.)</td>
<td></td>
<td>• Instance 3-Instance 4</td>
<td></td>
</tr>
<tr>
<td>32:</td>
<td>2</td>
<td>4:</td>
<td>25%</td>
</tr>
<tr>
<td>• 8</td>
<td></td>
<td>• Instance 1</td>
<td></td>
</tr>
<tr>
<td>• 8</td>
<td></td>
<td>• Instance 2</td>
<td></td>
</tr>
<tr>
<td>• 8</td>
<td></td>
<td>• Instance 3</td>
<td></td>
</tr>
<tr>
<td>• 8</td>
<td></td>
<td>• Instance 4</td>
<td></td>
</tr>
</tbody>
</table>
The following table applies to the Firepower 9300 with one SM-44 using subinterfaces on a single parent physical interface. For example, create a large EtherChannel to bundle all of your like-kind interfaces together, and then share subinterfaces of that EtherChannel. Sharing multiple physical interfaces uses more forwarding table resources than sharing multiple subinterfaces.

The Firepower 9300 with one SM-44 can support up to 14 instances.

### Table 5: Subinterfaces on One Parent and Instances on a Firepower 9300 with One SM-44

<table>
<thead>
<tr>
<th>Dedicated Interfaces</th>
<th>Shared Interfaces</th>
<th>Number of Instances</th>
<th>% Forwarding Table Used</th>
</tr>
</thead>
<tbody>
<tr>
<td>112:</td>
<td>0</td>
<td>14:</td>
<td>17%</td>
</tr>
<tr>
<td>• 112 (8 ea.)</td>
<td></td>
<td>• Instance 1-Instance 14</td>
<td></td>
</tr>
<tr>
<td>224:</td>
<td>0</td>
<td>14:</td>
<td>17%</td>
</tr>
<tr>
<td>• 224 (16 ea.)</td>
<td></td>
<td>• Instance 1-Instance 14</td>
<td></td>
</tr>
<tr>
<td>14:</td>
<td>1</td>
<td>14:</td>
<td>46%</td>
</tr>
<tr>
<td>• 14 (1 ea.)</td>
<td></td>
<td>• Instance 1-Instance 14</td>
<td></td>
</tr>
</tbody>
</table>
## Viewing Shared Interface Resources

To view forwarding table and VLAN group usage, see the **Instances > Interface Forwarding Utilization** area. For example:

<table>
<thead>
<tr>
<th>Dedicated Subinterfaces</th>
<th>Shared Subinterfaces</th>
<th>Number of Instances</th>
<th>% Forwarding Table Used</th>
</tr>
</thead>
<tbody>
<tr>
<td>14:</td>
<td>2:</td>
<td>14:</td>
<td></td>
</tr>
<tr>
<td>• 7 (1 ea.)</td>
<td>• 1</td>
<td>• Instance 1-Instance 7</td>
<td>37%</td>
</tr>
<tr>
<td>• 7 (1 ea.)</td>
<td>• 1</td>
<td>• Instance 8-Instance 14</td>
<td></td>
</tr>
<tr>
<td>112:</td>
<td>1</td>
<td>14:</td>
<td></td>
</tr>
<tr>
<td>• 112 (8 ea.)</td>
<td></td>
<td>• Instance 1-Instance 14</td>
<td>46%</td>
</tr>
<tr>
<td>112:</td>
<td>2:</td>
<td>14:</td>
<td></td>
</tr>
<tr>
<td>• 56 (8 ea.)</td>
<td>• 1</td>
<td>• Instance 1-Instance 7</td>
<td>37%</td>
</tr>
<tr>
<td>• 56 (8 ea.)</td>
<td>• 1</td>
<td>• Instance 8-Instance 14</td>
<td></td>
</tr>
<tr>
<td>112:</td>
<td>2</td>
<td>14:</td>
<td></td>
</tr>
<tr>
<td>• 112 (8 ea.)</td>
<td></td>
<td>• Instance 1-Instance 14</td>
<td>46%</td>
</tr>
<tr>
<td>112:</td>
<td>4:</td>
<td>14:</td>
<td></td>
</tr>
<tr>
<td>• 56 (8 ea.)</td>
<td>• 2</td>
<td>• Instance 1-Instance 7</td>
<td>37%</td>
</tr>
<tr>
<td>• 56 (8 ea.)</td>
<td>• 2</td>
<td>• Instance 8-Instance 14</td>
<td></td>
</tr>
<tr>
<td>140:</td>
<td>10</td>
<td>14:</td>
<td></td>
</tr>
<tr>
<td>• 140 (10 ea.)</td>
<td></td>
<td>• Instance 1-Instance 14</td>
<td>46%</td>
</tr>
<tr>
<td>140:</td>
<td>20:</td>
<td>14:</td>
<td></td>
</tr>
<tr>
<td>• 70 (10 ea.)</td>
<td>• 10</td>
<td>• Instance 1-Instance 7</td>
<td>37%</td>
</tr>
<tr>
<td>• 70 (10 ea.)</td>
<td>• 10</td>
<td>• Instance 8-Instance 14</td>
<td></td>
</tr>
</tbody>
</table>
How the Chassis Classifies Packets

Each packet that enters the chassis must be classified, so that the chassis can determine to which instance to send a packet.

- Unique Interfaces—If only one instance is associated with the ingress interface, the chassis classifies the packet into that instance. For bridge group member interfaces (in transparent mode or routed mode), inline sets, or passive interfaces, this method is used to classify packets at all times.

- Unique MAC Addresses—The chassis automatically generates unique MAC addresses for all interfaces, including shared interfaces. If multiple instances share an interface, then the classifier uses unique MAC addresses assigned to the interface in each instance. An upstream router cannot route directly to an instance without unique MAC addresses. You can also set the MAC addresses manually when you configure each interface within the application.

If the destination MAC address is a multicast or broadcast MAC address, the packet is duplicated and delivered to each instance.

Classification Examples

The following figure shows multiple instances sharing an outside interface. The classifier assigns the packet to Instance C because Instance C includes the MAC address to which the router sends the packet.
Figure 5: Packet Classification with a Shared Interface Using MAC Addresses

Note that all new incoming traffic must be classified, even from inside networks. The following figure shows a host on the Instance C inside network accessing the internet. The classifier assigns the packet to Instance C because the ingress interface is Ethernet 1/2.3, which is assigned to Instance C.
For transparent firewalls, you must use unique interfaces. The following figure shows a packet destined to a host on the Instance C inside network from the internet. The classifier assigns the packet to Instance C because the ingress interface is Ethernet 1/2.3, which is assigned to Instance C.
For inline sets, you must use unique interfaces and they must be physical interfaces or EtherChannels. The following figure shows a packet destined to a host on the Instance C inside network from the internet. The classifier assigns the packet to Instance C because the ingress interface is Ethernet 1/5, which is assigned to Instance C.
Cascading Container Instances

Placing a container instance directly in front of another instance is called *cascading container instances*; the outside interface of one instance is the same interface as the inside interface of another instance. You might want to cascade instances if you want to simplify the configuration of some instances by configuring shared parameters in the top instance.

The following figure shows a gateway instance with two instances behind the gateway.
The following example includes three container instances in routed firewall mode. They include the following interfaces:

- **Management**—All instances use the Port-Channel1 interface (management type). This EtherChannel includes two 10 Gigabit Ethernet interfaces. Within each application, the interface uses a unique IP address on the same management network.

- **Inside**—Each instance uses a subinterface on Port-Channel2 (data type). This EtherChannel includes two 10 Gigabit Ethernet interfaces. Each subinterface is on a separate network.

- **Outside**—All instances use the Port-Channel3 interface (data-sharing type). This EtherChannel includes two 10 Gigabit Ethernet interfaces. Within each application, the interface uses a unique IP address on the same outside network.

- **Failover**—Each instance uses a subinterface on Port-Channel4 (data type). This EtherChannel includes two 10 Gigabit Ethernet interfaces. Each subinterface is on a separate network.
Automatic MAC Addresses for Container Instance Interfaces

The FXOS chassis automatically generates MAC addresses for container instance interfaces, and guarantees that a shared interface in each instance uses a unique MAC address.

If you manually assign a MAC address to a shared interface within the application, then the manually-assigned MAC address is used. If you later remove the manual MAC address, the autogenerated address is used. In the rare circumstance that the generated MAC address conflicts with another private MAC address in your network, we suggest that you manually set the MAC address for the interface within the application.

Because autogenerated addresses start with A2, you should not start manual MAC addresses with A2 due to the risk of overlapping addresses.

The FXOS chassis generates the MAC address using the following format:

A2xx.yyzz.zzzz

Where xx.yy is a user-defined prefix or a system-defined prefix, and zz.zzzz is an internal counter generated by the chassis. The system-defined prefix matches the lower 2 bytes of the first MAC address in the burned-in MAC address pool that is programmed into the IDPROM. Use connect fxos, then show module to view the MAC address pool. For example, if the range of MAC addresses shown for module 1 is b0aa.772f.f0b0 to b0aa.772f.f0bf, then the system prefix will be f0b0.

The user-defined prefix is an integer that is converted into hexadecimal. For an example of how the user-defined prefix is used, if you set a prefix of 77, then the chassis converts 77 into the hexadecimal value 004D (yyxx).

When used in the MAC address, the prefix is reversed (xxyy) to match the chassis native form:

A24D.00zz.zzzz

For a prefix of 0009 (03F1), the MAC address is:

A2F1.03zz.zzzz
Container Instance Resource Management

To specify resource usage per container instance, create one or more resource profiles in FXOS. When you deploy the logical device/application instance, you specify the resource profile that you want to use. The resource profile sets the number of CPU cores; RAM is dynamically allocated according to the number of cores, and disk space is set to 40 GB per instance. To view the available resources per model, see Requirements and Prerequisites for Container Instances, on page 21. To add a resource profile, see Add a Resource Profile for Container Instances, on page 25.

Performance Scaling Factor for Multi-Instance Capability

The maximum throughput (connections, VPN sessions, and TLS proxy sessions) for a platform is calculated for a native instance’s use of memory and CPU (and this value is shown in show resource usage). If you use multiple instances, then you need to calculate the throughput based on the percentage of CPU cores that you assign to the instance. For example, if you use a container instance with 50% of the cores, then you should initially calculate 50% of the throughput. Moreover, the throughput available to a container instance may be less than that available to a native instance.


Container Instances and High Availability

You can use High Availability using a container instance on 2 separate chassis; for example, if you have 2 chassis, each with 10 instances, you can create 10 High Availability pairs. Note that High Availability is not configured in FXOS; configure each High Availability pair in the application manager.

For detailed requirements, see Requirements and Prerequisites for High Availability, on page 22 and Add a High Availability Pair, on page 36.

End-to-End Procedure

This procedure configures all necessary elements for a multi-instance environment.

Procedure

Step 1  Add a Resource Profile for Container Instances, on page 25.

The chassis includes a default resource profile called "Default-Small," which includes the minimum number of cores. If you do not want to use only this profile, you need to add profiles before you add the container instances.

Step 2  (Optional) Add a MAC Pool Prefix and View MAC Addresses for Container Instance Interfaces, on page 26.
The FXOS chassis automatically generates MAC addresses for container instance interfaces, and guarantees that a shared interface in each instance uses a unique MAC address. You can optionally define the prefix used in generation.

**Step 3**  Configure Interfaces, on page 27.

You can use interfaces with container instances the same way as you would with native instances. However, VLAN subinterfaces and data-shared interfaces are only available for container instances and provide scalability and flexibility for your deployment. Be sure to read about limitations on shared interfaces in this guide.

**Step 4**  Add a Standalone Firepower Threat Defense for FMC, on page 31.

Container instances are only supported for FTD as a standalone device or a failover pair; clustering is not supported.

**Step 5**  Add a High Availability Pair, on page 36.

If you want to deploy a High Availability pair, refer to the requirements in this section.

---

**Requirements and Prerequisites for Container Instances**

**Supported Application Types**

- Firepower Threat Defense using FMC

**Maximum Container Instances and Resources per Model**

For each container instance, you can specify the number of CPU cores to assign to the instance. RAM is dynamically allocated according to the number of cores, and disk space is set to 40 GB per instance.

<table>
<thead>
<tr>
<th>Model</th>
<th>Max. Container Instances</th>
<th>Available CPU Cores</th>
<th>Available RAM</th>
<th>Available Disk Space</th>
</tr>
</thead>
<tbody>
<tr>
<td>Firepower 4110</td>
<td>3</td>
<td>22</td>
<td>53 GB</td>
<td>125.6 GB</td>
</tr>
<tr>
<td>Firepower 4112</td>
<td>3</td>
<td>22</td>
<td>78 GB</td>
<td>308 GB</td>
</tr>
<tr>
<td>Firepower 4115</td>
<td>7</td>
<td>46</td>
<td>162 GB</td>
<td>308 GB</td>
</tr>
<tr>
<td>Firepower 4120</td>
<td>3</td>
<td>46</td>
<td>101 GB</td>
<td>125.6 GB</td>
</tr>
<tr>
<td>Firepower 4125</td>
<td>10</td>
<td>62</td>
<td>162 GB</td>
<td>644 GB</td>
</tr>
<tr>
<td>Firepower 4140</td>
<td>7</td>
<td>70</td>
<td>222 GB</td>
<td>311.8 GB</td>
</tr>
<tr>
<td>Firepower 4145</td>
<td>14</td>
<td>86</td>
<td>344 GB</td>
<td>608 GB</td>
</tr>
<tr>
<td>Firepower 4150</td>
<td>7</td>
<td>86</td>
<td>222 GB</td>
<td>311.8 GB</td>
</tr>
<tr>
<td>Model</td>
<td>Max. Container Instances</td>
<td>Available CPU Cores</td>
<td>Available RAM</td>
<td>Available Disk Space</td>
</tr>
<tr>
<td>--------------------------------------------</td>
<td>--------------------------</td>
<td>---------------------</td>
<td>---------------</td>
<td>---------------------</td>
</tr>
<tr>
<td>Firepower 9300 SM-24 security module</td>
<td>7</td>
<td>46</td>
<td>226 GB</td>
<td>656.4 GB</td>
</tr>
<tr>
<td>Firepower 9300 SM-36 security module</td>
<td>11</td>
<td>70</td>
<td>222 GB</td>
<td>640.4 GB</td>
</tr>
<tr>
<td>Firepower 9300 SM-40 security module</td>
<td>13</td>
<td>78</td>
<td>334 GB</td>
<td>1359 GB</td>
</tr>
<tr>
<td>Firepower 9300 SM-44 security module</td>
<td>14</td>
<td>86</td>
<td>218 GB</td>
<td>628.4 GB</td>
</tr>
<tr>
<td>Firepower 9300 SM-48 security module</td>
<td>15</td>
<td>94</td>
<td>334 GB</td>
<td>1341 GB</td>
</tr>
<tr>
<td>Firepower 9300 SM-56 security module</td>
<td>18</td>
<td>110</td>
<td>334 GB</td>
<td>1314 GB</td>
</tr>
</tbody>
</table>

**Firepower Management Center Requirements**

For all instances on a Firepower 4100 chassis or Firepower 9300 module, you must use the same Firepower Management Center (FMC) due to the licensing implementation.

**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.

- High Availability is only supported between same-type modules on the Firepower 9300; but the two chassis can include mixed modules. For example, each chassis has an SM-36, SM-40, and SM-44. You can create High Availability pairs between the SM-36 modules, between the SM-40 modules, and between the SM-44 modules.

- For container instances, each unit must use the same resource profile attributes.

- For other High Availability system requirements, see the application configuration guide chapter for High Availability.
Guidelines and Limitations

General Guidelines

• Multi-instance capability with container instances is only available for the FTD using FMC.

• For FTD container instances, a single Firepower Management Center must manage all instances on a security module/engine.

• You can enable TLS crypto acceleration on up to 16 container instances.

• For FTD container instances, the following features are not supported:
  • Radware DefensePro link decorator
  • FMC UCAPL/CC mode
  • Flow offload to hardware

VLAN Subinterfaces

• Subinterfaces (and the parent interfaces) can only be assigned to container instances.

  Note
  If you assign a parent interface to a container instance, it only passes untagged (non-VLAN) traffic. Do not assign the parent interface unless you intend to pass untagged traffic. For Cluster type interfaces, the parent interface cannot be used.

• Subinterfaces are supported on Data or Data-sharing type interfaces, as well as Cluster type interfaces. If you add subinterfaces to a Cluster interface, you cannot use that interface for a native cluster.

• For multi-instance clustering, subinterfaces are not supported on Data interfaces. However, subinterfaces are supported for the cluster control link, so you can use either a dedicated EtherChannel or a subinterface of an EtherChannel for the cluster control link.

• You can create up to 500 VLAN IDs.

• See the following limitations within the logical device application; keep these limitations in mind when planning your interface allocation.
  • You cannot use subinterfaces for an FTD inline set or as a passive interface.
  • If you use a subinterface for the failover link, then all subinterfaces on that parent, and the parent itself, are restricted for use as failover links. You cannot use some subinterfaces as failover links, and some as regular data interfaces.

Data-sharing Interfaces

• You cannot use a data-sharing interface with a native instance.

• Maximum 14 instances per shared interface. For example, you can allocate Ethernet1/1 to Instance1 through Instance14.
Maximum 10 shared interfaces per instance. For example, you can allocate Ethernet1/1.1 through Ethernet1/1.10 to Instance1.

- You cannot use a data-sharing interface in a cluster.

- See the following limitations within the logical device application; keep these limitations in mind when planning your interface allocation.
  - You cannot use a data-sharing interface with a transparent firewall mode device.
  - You cannot use a data-sharing interface with FTD inline sets or passive interfaces.
  - You cannot use a data-sharing interface for the failover link.

**Default MAC Addresses**

- MAC addresses for all interfaces are taken from a MAC address pool. For subinterfaces, if you decide to manually configure MAC addresses, make sure you use unique MAC addresses for all subinterfaces on the same parent interface to ensure proper classification. See Automatic MAC Addresses for Container Instance Interfaces, on page 19.
Add a Resource Profile for Container Instances

To specify resource usage per container instance, create one or more resource profiles. When you deploy the logical device/application instance, you specify the resource profile that you want to use. The resource profile sets the number of CPU cores; RAM is dynamically allocated according to the number of cores, and disk space is set to 40 GB per instance.

- The minimum number of cores is 6.

Note

Instances with a smaller number of cores might experience relatively higher CPU utilization than those with larger numbers of cores. Instances with a smaller number of cores are more sensitive to traffic load changes. If you experience traffic drops, try assigning more cores.

- You can assign cores as an even number (6, 8, 10, 12, 14 etc.) up to the maximum.

- The maximum number of cores available depends on the security module/chassis model; see Requirements and Prerequisites for Container Instances, on page 21.

The chassis includes a default resource profile called "Default-Small," which includes the minimum number of cores. You can change the definition of this profile, and even delete it if it is not in use. Note that this profile is created when the chassis reloads and no other profile exists on the system.

You cannot change the resource profile settings if it is currently in use. You must disable any instances that use it, then change the resource profile, and finally reenable the instance. If you resize instances in an established High Availability pair or cluster, then you should make all members the same size as soon as possible.

If you change the resource profile settings after you add the FTD instance to the FMC, then update the inventory for each unit on the FMC Devices > Device Management > Device > System > Inventory dialog box.

Procedure

Step 1  Choose Platform Settings > Resource Profiles, and click Add.

The Add Resource Profile dialog box appears.

Step 2  Set the following parameters.

- **Name**—Sets the name of the profile between 1 and 64 characters. Note that you cannot change the name of this profile after you add it.

- **Description**—Sets the description of the profile up to 510 characters.

- **Number of Cores**—Sets the number of cores for the profile, between 6 and the maximum, depending on your chassis, as an even number.

Step 3  Click OK.
Add a MAC Pool Prefix and View MAC Addresses for Container Instance Interfaces

The FXOS chassis automatically generates MAC addresses for container instance interfaces, and guarantees that a shared interface in each instance uses a unique MAC address. The FXOS chassis generates the MAC address using the following format:

A2xx.yyzz.zzzz

Where xx.yy is a user-defined prefix or a system-defined prefix, and zz.zzzz is an internal counter generated by the chassis. The system-defined prefix matches the lower 2 bytes of the first MAC address in the burned-in MAC address pool that is programmed into the IDPROM. Use `connect fxos`, then `show module` to view the MAC address pool. For example, if the range of MAC addresses shown for module 1 is b0aa.772f.f0b0 to b0aa.772f.f0bf, then the system prefix will be f0b0.

See Automatic MAC Addresses for Container Instance Interfaces, on page 19 for more information.

This procedure describes how to view the MAC addresses and how to optionally define the prefix used in generation.

**Note**

If you change the MAC address prefix after you deploy logical devices, you may experience traffic interruption.

**Procedure**

**Step 1**

Choose **Platform Settings > MAC Pool**.

This page shows generated MAC addresses along with the container instance and interface using the MAC address.

**Step 2**

(Optional) Add a MAC address prefix used in generating the MAC addresses.

a) Click **Add Prefix**.

   The **Set the Prefix for the MAC Pool** dialogue box appears.

a) Enter a decimal value between 1 and 65535. This prefix is converted to a four-digit hexadecimal number, and used as part of the MAC address.

   For an example of how the prefix is used, if you set a prefix of 77, then the chassis converts 77 into the hexadecimal value 004D (yyxx). When used in the MAC address, the prefix is reversed (xxyy) to match the chassis native form:

   A24D.00zz.zzzz

   For a prefix of 1009 (03F1), the MAC address is:

   A2F1.03zz.zzzz

b) Click **OK**.
New MAC addresses using the prefix are generated and assigned. The current prefix and the resulting hex value display above the table.

Configure Interfaces

By default, physical interfaces are disabled. You can enable interfaces, add EtherChannels, add VLAN subinterfaces, edit interface properties, and configure breakout ports.

Note

If you remove an interface in FXOS (for example, if you remove a network module, remove an EtherChannel, or reassign an interface to an EtherChannel), then the ASA configuration retains the original commands so that you can make any necessary adjustments; removing an interface from the configuration can have wide effects. You can manually remove the old interface configuration in the ASA OS.

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
Choose Interfaces to open the Interfaces page.

The All Interfaces page shows a visual representation of the currently installed interfaces at the top of the page and provides a listing of the installed interfaces in the table below.

Step 2
Click Edit in the row for the interface you want to edit to open the Edit Interface dialog box.

Step 3
To enable the interface, check the Enable check box. To disable the interface, uncheck the Enable check box.

Step 4
Choose the interface Type:

- Data
- Data-sharing—For container instances only.
- Mgmt
- Firepower-eventing—For FTD only.
- Cluster—Do not choose the Cluster type; by default, the cluster control link is automatically created on Port-channel 48.
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.

You can configure each physical Data or Data-sharing interface in an EtherChannel to be:

- Active—Sends and receives LACP updates. An active EtherChannel can establish connectivity with either an active or a passive EtherChannel. You should use the active mode unless you need to minimize the amount of LACP traffic.

- On—The EtherChannel is always on, and LACP is not used. An “on” EtherChannel can only establish a connection with another “on” EtherChannel.

It may take up to three minutes for an EtherChannel to come up to an operational state if you change its mode from On to Active or from Active to On.

Non-data interfaces only support active mode.

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. “On” mode cannot use standby interfaces in the channel group when an interface goes down, and the connectivity and configurations are not checked.

When the Firepower 4100/9300 chassis creates an EtherChannel, the EtherChannel stays in a Suspended state for Active LACP mode or a Down state for On LACP mode 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 or Down state.
Procedure

Step 1  Choose **Interfaces** to open the Interfaces page.

The **All Interfaces** page shows a visual representation of the currently installed interfaces at the top of the page and provides a listing of the installed interfaces in the table below.

Step 2  Click **Add Port Channel** above the interfaces table to open the **Add Port Channel** dialog box.

Step 3  Enter an ID for the port channel in the **Port Channel ID** field. Valid values are between 1 and 47.

Port-channel 48 is reserved for the cluster control link when you deploy a clustered logical device. If you do not want to use Port-channel 48 for the cluster control link, you can delete it and configure a Cluster type EtherChannel with a different ID. You can add multiple Cluster type EtherChannels and add VLAN subinterfaces for use with multi-instance clustering. For intra-chassis clustering, do not assign any interfaces to the Cluster EtherChannel.

Step 4  To enable the port channel, check the **Enable** check box. To disable the port channel, uncheck the **Enable** check box.

Step 5  Choose the interface **Type**:

- **Data**
- **Data-sharing**—For container instances only.
- **Mgmt**
- **Firepower-eventing**—For FTD only.
- **Cluster**

Step 6  Set the required **Admin Speed** for the member interfaces from the drop-down list.

If you add a member interface that is not at the specified speed, it will not successfully join the port channel.

Step 7  For Data or Data-sharing interfaces, choose the LACP port-channel **Mode**, **Active** or **On**.

For non-Data or non-Data-sharing interfaces, the mode is always active.

Step 8  Set the required **Admin Duplex** for the member interfaces, **Full Duplex** or **Half Duplex**.

If you add a member interface that is configured with the specified duplex, it will not successfully join the port channel.

Step 9  To add an interface to the port channel, select the interface in the **Available Interface** list and click **Add Interface** to move the interface to the **Member ID** list.

You can add up to 16 member interfaces of the same media type and capacity. The member interfaces must be set to the same speed and duplex, and must match the speed and duplex that you configured for this port channel. 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.

**Tip**  You can add multiple interfaces at one time. To select multiple individual interfaces, click on the desired interfaces while holding down the **Ctrl** key. To select a range of interfaces, select the first interface in the range, and then, while holding down the **Shift** key, click to select the last interface in the range.
Add a VLAN Subinterface for Container Instances

You can add up to 500 subinterfaces to your chassis.

For multi-instance clustering, you can only add subinterfaces to the Cluster-type interface; subinterfaces on data interfaces are not supported.

VLAN IDs per interface must be unique, and within a container instance, VLAN IDs must be unique across all assigned interfaces. You can reuse VLAN IDs on separate interfaces as long as they are assigned to different container instances. However, each subinterface still counts towards the limit even though it uses the same ID.

You can also add subinterfaces within the application.

Procedure

Step 1  Choose Interfaces to open the All Interfaces tab.

The All Interfaces tab shows a visual representation of the currently installed interfaces at the top of the page and provides a listing of the installed interfaces in the table below.

Step 2  Click Add New > Subinterface to open the Add Subinterface dialog box.

Step 3  Choose the interface Type:

- Data
- Data-sharing
- Cluster—If you add subinterfaces to a Cluster interface, you cannot use that interface for a native cluster.

For Data and Data-sharing interfaces: The type is independent of the parent interface type; you can have a Data-sharing parent and a Data subinterface, for example.

Step 4  Choose the parent Interface from the drop-down list.

You cannot add a subinterface to a physical interface that is currently allocated to a logical device. If other subinterfaces of the parent are allocated, you can add a new subinterface as long as the parent interface itself is not allocated.

Step 5  Enter a Subinterface ID, between 1 and 4294967295.

This ID will be appended to the parent interface ID as interface_id.subinterface_id. For example, if you add a subinterface to Ethernet1/1 with the ID of 100, then the subinterface ID will be: Ethernet1/1.100. This ID is not the same as the VLAN ID, although you can set them to match for convenience.

Step 6  Set the VLAN ID between 1 and 4095.

Step 7  Click OK.
Expand the parent interface to view all subinterfaces under it.

---

### Add a Standalone Firepower Threat Defense for FMC

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 use native instances on some modules, and container instances on the other module(s).

**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.

  **Note** For the Firepower 9300, you can install different application types (ASA and FTD) on separate modules in the chassis. You can also run different versions of an application instance type on separate modules.

- 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 (and that appears at the top of the Interfaces tab as MGMT).

- 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. See the `configure network management-data-interface` command in the FTD command reference for more information.

- 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 2 for more information.

- For container instances, if you do not want to use the default profile, add a resource profile according to Add a Resource Profile for Container Instances, on page 25.

- For container instances, before you can install a container instance for the first time, you must reinitialize the security module/engine so that the disk has the correct formatting. Choose Security Modules or Security Engine, and click the Reinitialize icon. An existing logical device will be deleted and then reinstalled as a new device, losing any local application configuration. If you are replacing a native instance with container instances, you will need to delete the native instance in any case. You cannot automatically migrate a native instance to a container instance.

- 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** Choose **Logical Devices**.

**Step 2** Click **Add > Standalone**, and set the following parameters:

![Add Standalone](image)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Device Name</strong></td>
<td>FTD_Instance2</td>
</tr>
<tr>
<td><strong>Template</strong></td>
<td>Cisco Firepower Threat Defense</td>
</tr>
<tr>
<td><strong>Image Version</strong></td>
<td>6.5.0.1159</td>
</tr>
<tr>
<td><strong>Instance Type</strong></td>
<td>Container</td>
</tr>
</tbody>
</table>

- **a)** Provide a **Device Name**.
  This name is used by the chassis supervisor to configure management settings and to assign interfaces; it is not the device name used in the application configuration.

- **b)** For the **Template**, choose **Cisco Firepower Threat Defense**.

- **c)** Choose the **Image Version**.

- **d)** Choose the **Instance Type**: **Container** or **Native**.
  A native instance uses all of the resources (CPU, RAM, and disk space) of the security module/engine, so you can only install one native instance. A container instance uses a subset of resources of the security module/engine, so you can install multiple container instances.

- **e)** Click **OK**.
  You see the Provisioning - *device name* window.

**Step 3** Expand the **Data Ports** area, and click each interface that you want to assign to the device.
You can only assign data and data-sharing interfaces that you previously enabled on the Interfaces page. You will later enable and configure these interfaces in FMC, including setting the IP addresses.

You can only assign up to 10 data-sharing interfaces to a container instance. Also, each data-sharing interface can be assigned to at most 14 container instances. A data-sharing interface is indicated by the sharing icon (🔗).

Hardware Bypass-capable ports are shown with the following icon: 🔒. For certain interface modules, you can enable the Hardware Bypass feature for Inline Set interfaces only (see the FMC configuration guide). Hardware Bypass ensures that traffic continues to flow between an inline interface pair during a power outage. This feature can be used to maintain network connectivity in the case of software or hardware failures. If you do not assign both interfaces in a Hardware Bypass pair, you see a warning message to make sure your assignment is intentional. You do not need to use the Hardware Bypass feature, so you can assign single interfaces if you prefer.

**Step 4**

Click the device icon in the center of the screen.

A dialog box appears where you can configure initial bootstrap settings. 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.

**Step 5**

On the General Information page, complete the following:
a) (For the Firepower 9300) Under **Security Module Selection** click the security module that you want to use for this logical device.

b) For a container instance, specify the **Resource Profile**.

   If you later assign a different resource profile, then the instance will reload, which can take approximately 5 minutes. Note that for established High Availability pairs, if you assign a different-sized resource profile, be sure to make all members the same size as soon as possible.

c) Choose the **Management Interface**.

   This interface is used to manage the logical device. This interface is separate from the chassis management port.

d) Choose the management interface **Address Type**: IPv4 only, IPv6 only, or IPv4 and IPv6.

e) Configure the **Management IP** address.

   Set a unique IP address for this interface.

   f) Enter a **Network Mask** or **Prefix Length**.

g) Enter a **Network Gateway** address.

---

**Step 6**

On the **Settings** tab, complete the following:
a) For a native instance, in the **Management type of application instance** drop-down list, choose FMC. Native instances also support FDM as a manager. After you deploy the logical device, you cannot change the manager type.

b) Enter the **Firepower Management Center IP** of the managing FMC. If you do not know the FMC IP address, leave this field blank and enter a passphrase in the **Firepower Management Center NAT ID** field.

c) For a container instance, **Permit Expert mode from FTD SSH sessions**: Yes or No. Expert Mode provides FTD shell access for advanced troubleshooting.
   
   If you choose Yes for this option, then users who access the container instance directly from an SSH session can enter Expert Mode. If you choose No, then only users who access the container instance from the FXOS CLI can enter Expert Mode. We recommend choosing No to increase isolation between instances.

   Use Expert Mode only if a documented procedure tells you it is required, or if the Cisco Technical Assistance Center asks you to use it. To enter this mode, use the `expert` command in the FTD CLI.

d) Enter the **Search Domains** as a comma-separated list.

e) Choose the **Firewall Mode**: Transparent or Routed.

   In routed mode, the FTD 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.

f) Enter the **DNS Servers** as a comma-separated list.

   The FTD uses DNS if you specify a hostname for the FMC, for example.

g) Enter the **Fully Qualified Hostname** for the FTD.

h) Enter a **Registration Key** to be shared between the FMC and the device during registration.
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.

i) Enter a **Password** for the FTD admin user for CLI access.

j) Choose the **Eventing Interface** on which Firepower events should be sent. If not specified, the management interface will be used.

   This interface must be defined as a Firepower-eventing interface.

k) For a container instance, set the **Hardware Crypto** as **Enabled** or **Disabled**.

   This setting enables TLS crypto acceleration in hardware, and improves performance for certain types of traffic. This feature is enabled by default. You can enable TLS crypto acceleration for up to 16 instances per security module. This feature is always enabled for native instances. To view the percentage of hardware crypto resources allocated to this instance, enter the `show hw-crypto` command.

---

**Step 7**

On the **Agreement** tab, read and accept the end user license agreement (EULA).

**Step 8**

Click **OK** to close the configuration dialog box.

**Step 9**

Click **Save**.

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 **Logical Devices** page for the status of the new logical device. When the logical device shows its **Status** as **online**, you can start configuring the security policy in the application.

---

**Step 10**

See the FMC configuration guide to add the FTD as a managed device and start configuring your security policy.

---

**Add a High Availability Pair**

or ASA 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 **Requirements and Prerequisites for High Availability**, on page 22.
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.

For container instances, data-sharing interfaces are not supported for the failover link. We recommend that you create subinterfaces on a parent interface or EtherChannel, and assign a subinterface for each instance to use as a failover link. Note that you must use all subinterfaces on the same parent as failover links. You cannot use one subinterface as a failover link and then use other subinterfaces (or the parent interface) as regular data 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.

**Note** For the ASA, if you remove an interface in FXOS (for example, if you remove a network module, remove an EtherChannel, or reassign an interface to an EtherChannel), then the ASA configuration retains the original commands so that you can make any necessary adjustments; removing an interface from the configuration can have wide effects. You can manually remove the old interface configuration in the ASA OS.

Troubleshooting Interfaces

**Troubleshooting Interfaces**

Error: The Switch Forwarding Path has 1076 entries and exceeds the limit of 1024. If you are adding an interface, reduce the number of shared interfaces assigned to logical devices, reduce the number of logical devices sharing interfaces, or use non-shared subinterfaces instead. If you are deleting a subinterface, you are seeing this message because the remaining configuration is no longer optimized to fit within the Switch Forwarding Path table. See the FXOS configuration guide for troubleshooting information about the deletion use case. Use 'show detail' under scope 'fabric-interconnect' to view the current Switch Forwarding Path Entry Count.

If you see this error when trying to delete a shared subinterface from a logical device, it is because your new configuration is not following this guideline for shared subinterfaces: use the same set of subinterfaces with the same group of logical devices. If you delete a shared subinterface from one logical device, you can end up with more VLAN groups and therefore less efficient usage of the forwarding table. To work around this situation, you need to add and delete shared subinterfaces simultaneously using the CLI so that you maintain the same set of subinterfaces for the same group of logical devices.

See the following scenarios for more information. These scenarios start with the following interfaces and logical devices:

- Shared subinterface set on the same parent: Port-Channel1.100 (VLAN 100), Port-Channel1.200 (VLAN 200), Port-Channel1.300 (VLAN 300)
• Logical device group: LD1, LD2, LD3, and LD4

Scenario 1: Remove a subinterface from one logical device, but leave it assigned to other logical devices

Do not remove the subinterface. Instead, just disable it in the application configuration. If you have to remove the subinterface, you will need to reduce the number of shared interfaces in general to continue to fit in the forwarding table.

Scenario 2: Remove all subinterfaces in the set from one logical device

Remove all subinterfaces in the set from the logical device at the CLI, and then save the configuration so that the removal is simultaneous.

1. View the VLAN groups for reference. In the following output, group 1 includes VLAN 100, 200, and 300, representing the 3 shared subinterfaces.

   ```
   firepower# connect fxos
   [...]
   firepower(fxos)# show ingress-vlan-groups
   ID  Class ID  Status  INTF  Vlan Status
   1    1     configured 100  present
         200  present
         300  present
   2048 512  configured 0  present
   2049 511  configured 0  present
   firepower(fxos)# exit
   firepower#
   ```

2. View the shared subinterfaces assigned to the logical device you want to change.

   ```
   firepower# scope ssa
   firepower /ssa # scope logical-device LD1
   firepower /ssa/logical-device # show external-port-link
   External-Port Link:
   Name Port or Port Channel Name Port Type App Name
   Description
   ------------------------------ ------------------------------- --------------
   Ethernet14_ftd Ethernet1/4 Mgmt ftd
   PC1.100_ftd Port-channel1.100 Data Sharing ftd
   PC1.200_ftd Port-channel1.200 Data Sharing ftd
   PC1.300_ftd Port-channel1.300 Data Sharing ftd
   ```

3. Remove the subinterfaces from the logical device, and then save the configuration.

   ```
   firepower /ssa/logical-device # delete external-port-link PC1.100_ftd
   firepower /ssa/logical-device* # delete external-port-link PC1.200_ftd
   firepower /ssa/logical-device* # delete external-port-link PC1.300_ftd
   firepower /ssa/logical-device* # commit-buffer
   firepower /ssa/logical-device#
   ```

If you had committed the configuration in the middle, you would have ended up with 2 VLAN groups, which could have generated the switch forwarding path error and prevented you from saving the configuration.
Scenario 3: Remove a subinterface from all logical devices in the group

Remove the subinterface from all logical devices in the group at the CLI, and then save the configuration so that the removal is simultaneous. For example:

1. View the VLAN groups for reference. In the following output, group 1 includes VLAN 100, 200, and 300, representing the 3 shared subinterfaces.

```plaintext
firepower# connect fxos
[...]
firepower(fxos)# show ingress-vlan-groups
ID  Class ID  Status  INTF  Vlan Status
1   1        configured  100  present
     2        configured  200  present
     3        configured  300  present
2048 512 configured  0    present
2049 511 configured  0    present
```

2. View the interfaces assigned to each logical device, and note the shared subinterfaces in common. If they are on the same parent interface, they will belong to one VLAN group, and should match the `show ingress-vlan-groups` list. In Firepower Chassis Manager, you can hover over each shared subinterface to see which instances it is allocated to.

*Figure 10: Instances per shared interface*

At the CLI, you can view characteristics of all logical devices, including the allocated interfaces.

```plaintext
firepower# scope ssa
firepower /ssa # show logical-device expand

Logical Device:
Name: LD1
Description:
Slot ID: 1
Mode: Standalone
Oper State: Ok
Template Name: ftd

External-Port Link:
Name: Ethernet14_ftd
Port or Port Channel Name: Ethernet1/4
Port Type: Mgmt
App Name: ftd
Description:
```
Name: PC1.100_ftd
Port or Port Channel Name: Port-channel1.100
Port Type: Data Sharing
App Name: ftd
Description:

Name: PC1.200_ftd
Port or Port Channel Name: Port-channel1.200
Port Type: Data Sharing
App Name: ftd
Description:

System MAC address:
Mac Address
-------------
A2:F0:B0:00:00:25

Name: PC1.300_ftd
Port or Port Channel Name: Port-channel1.300
Port Type: Data Sharing
App Name: ftd
Description:

[...]

Name: LD2
Description:
Slot ID: 1
Mode: Standalone
Oper State: Ok
Template Name: ftd

External-Port Link:
Name: Ethernet14_ftd
Port or Port Channel Name: Ethernet1/4
Port Type: Mgmt
App Name: ftd
Description:

Name: PC1.100_ftd
Port or Port Channel Name: Port-channel1.100
Port Type: Data Sharing
App Name: ftd
Description:

Name: PC1.200_ftd
Port or Port Channel Name: Port-channel1.200
Port Type: Data Sharing
App Name: ftd
Description:

Name: PC1.300_ftd
Port or Port Channel Name: Port-channel1.300
Port Type: Data Sharing
App Name: ftd
Description:

System MAC address:
Mac Address
-------------
A2:F0:B0:00:00:28

Name: PC1.300_ftd
Port or Port Channel Name: Port-channel1.300
Port Type: Data Sharing
App Name: ftd
Description:

[...]
Name: LD3
Description:
Slot ID: 1
Mode: Standalone
Oper State: Ok
Template Name: ftd

External-Port Link:
Name: Ethernet14_ftd
Port or Port Channel Name: Ethernet1/4
Port Type: Mgmt
App Name: ftd
Description:

Name: PC1.100_ftd
Port or Port Channel Name: Port-channel1.100
Port Type: Data Sharing
App Name: ftd
Description:

Name: PC1.200_ftd
Port or Port Channel Name: Port-channel1.200
Port Type: Data Sharing
App Name: ftd
Description:

System MAC address:
Mac Address
-----------
A2:F0:B0:00:00:2B

Name: PC1.300_ftd
Port or Port Channel Name: Port-channel1.300
Port Type: Data Sharing
App Name: ftd
Description:

 [...]
3. Remove the subinterface from each logical device, and then save the configuration.

```plaintext
disable /ssa
firepower /ssa # scope logical device LD1
firepower /ssa/logical-device # delete external-port-link PC1.300_ftd
firepower /ssa/logical-device* # exit
firepower /ssa* # scope logical-device LD2
firepower /ssa/logical-device* # delete external-port-link PC1.300_ftd
firepower /ssa/logical-device* # exit
firepower /ssa* # scope logical-device LD3
firepower /ssa/logical-device* # delete external-port-link PC1.300_ftd
firepower /ssa/logical-device* # exit
firepower /ssa* # scope logical-device LD4
firepower /ssa/logical-device* # delete external-port-link PC1.300_ftd
firepower /ssa/logical-device* # commit-buffer
firepower /ssa/logical-device#
```

If you had committed the configuration in the middle, you would have ended up with 2 VLAN groups, which could have generated the switch forwarding path error and prevented you from saving the configuration.

**Scenario 4: Add a subinterface to one or more logical devices**

Add the subinterface to *all* logical devices in the group at the CLI, and then save the configuration so that the addition is simultaneous.

1. Add the subinterface to each logical device, and then save the configuration.

```plaintext
disable /ssa
firepower /ssa # scope ssa
firepower /ssa # scope logical-device LD1
firepower /ssa/logical-device # create external-port-link PC1.400_ftd Port-channel1.400 ftd
firepower /ssa/logical-device/external-port-link* # exit
firepower /ssa/logical-device* # exit
firepower /ssa # scope logical-device LD2
firepower /ssa/logical-device # create external-port-link PC1.400_ftd Port-channel1.400 ftd
firepower /ssa/logical-device/external-port-link* # exit
firepower /ssa/logical-device* # exit
firepower /ssa # scope logical-device LD3
firepower /ssa/logical-device # create external-port-link PC1.400_ftd Port-channel1.400 ftd
firepower /ssa/logical-device/external-port-link* # exit
firepower /ssa/logical-device* # exit
firepower /ssa # scope logical-device LD4
firepower /ssa/logical-device # create external-port-link PC1.400_ftd Port-channel1.400 ftd
firepower /ssa/logical-device/external-port-link* # exit
firepower /ssa/logical-device* # exit
```
firepower /ssa/logical-device/external-port-link* # commit-buffer
firepower /ssa/logical-device/external-port-link #

If you had committed the configuration in the middle, you would have ended up with 2 VLAN groups, which could have generated the switch forwarding path error and prevented you from saving the configuration.

2. You can check that the Port-channel1.400 VLAN ID was added to VLAN group 1.

firepower /ssa/logical-device/external-port-link # connect fxos
[...
firepower(fxos)# show ingress-vlan-groups
<table>
<thead>
<tr>
<th>ID</th>
<th>Class ID</th>
<th>Status</th>
<th>INTF Vlan Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>configured</td>
<td>200 present</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>100 present</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>300 present</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>400 present</td>
</tr>
<tr>
<td>2048</td>
<td>512</td>
<td>configured</td>
<td>0 present</td>
</tr>
<tr>
<td>2049</td>
<td>511</td>
<td>configured</td>
<td>0 present</td>
</tr>
</tbody>
</table>

firepower(fxos)# exit
firepower /ssa/logical-device/external-port-link #

History for Multi-Instance Capability

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>FTD configuration backup and restore using FMC for container instances</td>
<td>2.9.1</td>
<td>You can now use the FMC backup/restore tool on an FTD container instance.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>New/Modified FMC screens: <strong>System &gt; Tools &gt; Backup/Restore &gt; Managed Device Backup</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td>New/Modified FTD CLI commands: <strong>restore</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Supported platforms: Firepower 4100/9300</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Note</strong> Requires Firepower 6.7.</td>
</tr>
<tr>
<td>Feature Name</td>
<td>Platform Releases</td>
<td>Feature Information</td>
</tr>
<tr>
<td>------------------------------------------------</td>
<td>-------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| Multi-instance clustering                        | 2.8.1             | You can now create a cluster using container instances. On the Firepower 9300, you must include one container instance on each module in the cluster. You cannot add more than one container instance to the cluster per security engine/module. We recommend that you use the same security module or chassis model for each cluster instance. However, you can mix and match container instances on different Firepower 9300 security module types or Firepower 4100 models in the same cluster if required. You cannot mix Firepower 9300 and 4100 instances in the same cluster. New/modified screens:  
  - Logical Devices > Add Cluster  
  - Interfaces > All Interfaces > Add New drop-down menu > Subinterface > Type field  
  **Note** Requires Firepower 6.6. |
| TLS crypto acceleration for multiple container instances | 2.7.1             | TLS crypto acceleration is now supported on multiple container instances (up to 16) on a Firepower 4100/9300 chassis. Previously, you could enable TLS crypto acceleration for only one container instance per module/security engine. New instances have this feature enabled by default. However, the upgrade does not enable acceleration on existing instances. Instead, use the **enter hw-crypto** and then the **set admin-state enabled** FXOS commands. New/Modified Firepower Chassis Manager screens:  
  Logical Devices > Add Device > Settings > Hardware Crypto drop-down menu  
  **Note** Requires FTD 6.5 or later. |
| Firepower 4115, 4125, and 4145                    | 2.6.1             | We introduced the Firepower 4115, 4125, and 4145.  
  **Note** Requires Firepower 6.4.  
  No modified screens.                                                                                                                                       |
| Support for ASA and FTD on separate modules of the same Firepower 9300 | 2.6.1             | You can now deploy ASA and FTD logical devices on the same Firepower 9300.  
  **Note** Requires ASA 9.12(1) and Firepower 6.4.0.  
  No modified screens.                                                                                                                                       |
<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>For the FTD bootstrap configuration, you can now set the NAT ID for the FMC in the Firepower Chassis Manager</td>
<td>2.6.1</td>
<td>You can now set the FMC NAT ID in the Firepower Chassis Manager. Previously, you could only set the NAT ID within the FXOS CLI or FTD CLI. Normally, you need both IP addresses (along with a registration key) for both routing purposes and for authentication: the FMC specifies the device IP address, and the device specifies the FMC IP address. However, if you only know one of the IP addresses, which is the minimum requirement for routing purposes, then you must also specify a unique NAT ID on both sides of the connection to establish trust for the initial communication and to look up the correct registration key. The FMC and device use the registration key and NAT ID (instead of IP addresses) to authenticate and authorize for initial registration. New/Modified screens: Logical Devices &gt; Add Device &gt; Settings &gt; Firepower Management Center NAT ID field</td>
</tr>
<tr>
<td>Support for SSL hardware acceleration on one FTD container instance on a module/security engine</td>
<td>2.6.1</td>
<td>You can now enable SSL hardware acceleration for one container instance on a module/security engine. SSL hardware acceleration is disabled for other container instances, but enabled for native instances. See the Firepower Management Center configuration guide for more information. New/Modified commands: <code>config hwCrypto enable</code>, <code>show hwCrypto</code> No modified screens.</td>
</tr>
<tr>
<td>Feature Name</td>
<td>Platform Releases</td>
<td>Feature Information</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------------</td>
<td>---------------------</td>
</tr>
</tbody>
</table>
| Multi-instance capability for Firepower Threat Defense | 2.4.1 | You can now deploy multiple logical devices, each with a Firepower Threat Defense container instance, on a single security engine/module. Formerly, you could only deploy a single native application instance. Native instances are still also supported. For the Firepower 9300, you can use a native instance on some modules, and container instances on the other module(s).

To provide flexible physical interface use, you can create VLAN subinterfaces in FXOS and also share interfaces between multiple instances. When you deploy a container instance, you must specify the number of CPU cores assigned; RAM is dynamically allocated according to the number of cores, and disk space is set to 40 GB per instance. This resource management lets you customize performance capabilities for each instance.

You can use High Availability using a container instance on 2 separate chassis; for example, if you have 2 chassis, each with 10 instances, you can create 10 High Availability pairs. Clustering is not supported.

**Note**  Multi-instance capability is similar to ASA multiple context mode, although the implementation is different. Multiple context mode partitions a single application instance, while multi-instance capability allows independent container instances. Container instances allow hard resource separation, separate configuration management, separate reloads, separate software updates, and full Firepower Threat Defense feature support. Multiple context mode, due to shared resources, supports more contexts on a given platform. Multiple context mode is not available on the Firepower Threat Defense.

**Note**  Requires FTD Version 6.3 or later.

New/Modified Firepower Chassis Manager screens:

- **Overview > Devices**
- **Interfaces > All Interfaces > Add New** drop-down menu > **Subinterface**
- **Interfaces > All Interfaces > Type**
- **Logical Devices > Add Device**
- **Platform Settings > Mac Pool**
- **Platform Settings > Resource Profiles**

New/Modified Firepower Management Center screens:

- **Devices > Device Management > Edit** icon > **Interfaces** tab