Configure Secure Firewall 3100 FDM 7.7.0 Hardware Bypass

Contents

Introduction

This document describes how to configure Hardware Bypass for inline sets in Firepower Device Manager (FDM) managed Secure Firewall 7.7.0.

Prerequisites

Requirements

Cisco recommends that you have knowledge of these topics:

- Inline-sets
- Secure Firewall 3100 Series
- Firepower Device Manager Graphical User Interface (GUI)

Components Used

The information in this document is based on these software and hardware versions:

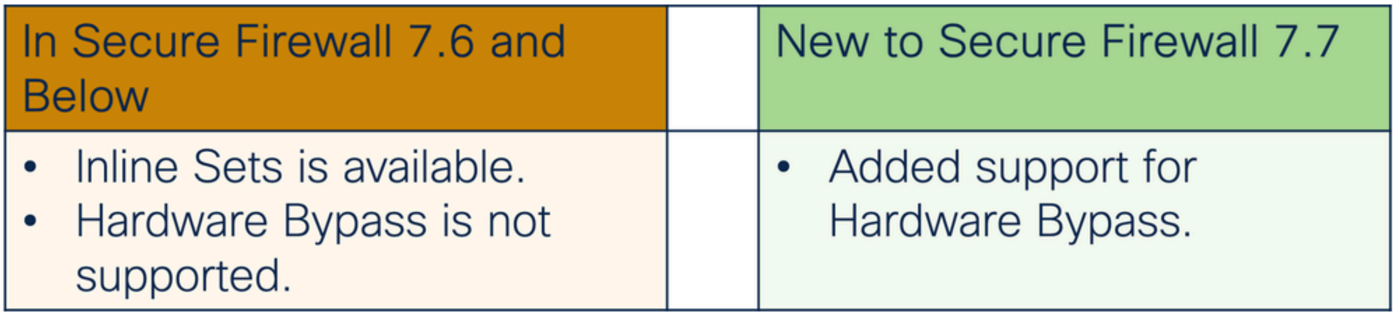
- Cisco Secure Firewall 3100 running v7.7.0.
- Cisco Secure Firewall Device Manager v7.7.0.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, ensure that you understand the potential impact of any command.

Background Information

Inline Sets feature was added to FDM in 7.4.1. Inline Sets enable inspection on an L2 network without the need of routing: Configure FTD Interfaces in Inline-Pair Mode

Contrasting Previous to This Release



In Secure Firewall 7.6 and Below		New to Secure Firewall 7.7
• Inline Sets is available. • Hardware Bypass is not supported.		• Added support for Hardware Bypass.

Secure Firewall 7.0 Bypass Feature

What’s New

- Hardware Inspection Bypass ensures that traffic continues to flow between an Inline Interface Pair during a power outage.
- This feature is used to maintain network connectivity in the case of software or hardware failures.
- Hardware Bypass is now available for Inline Sets for FDM 3100 Series platforms.

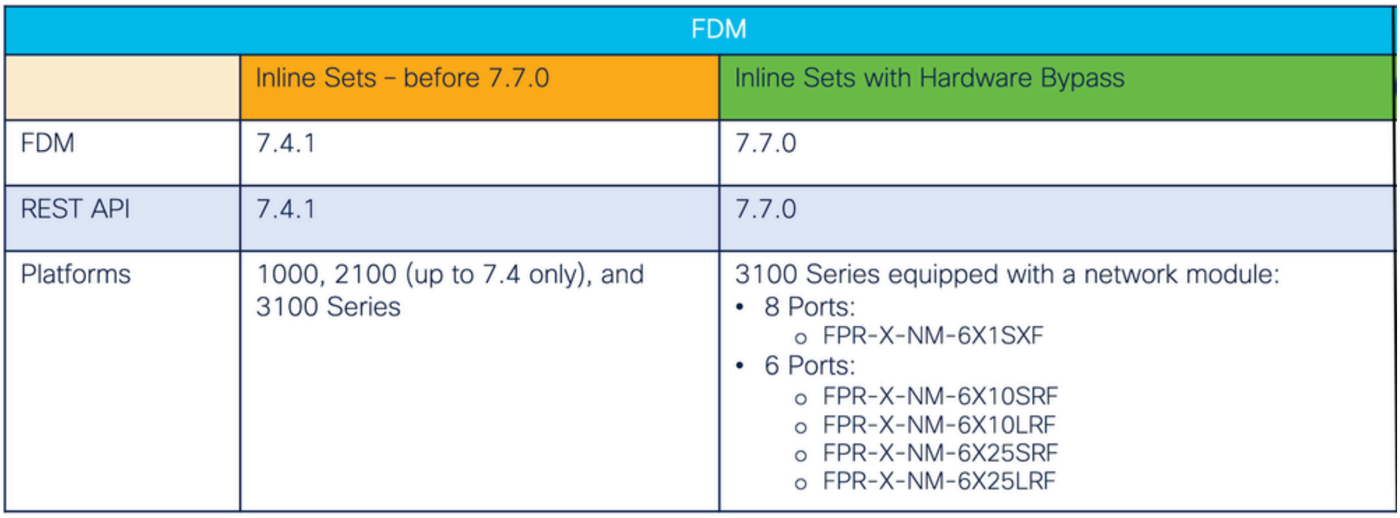
Note: Firepower Management Center Configuration Guide

Deployment Scenarios

- How would this feature fit into a production setup?
    - Inline sets are used for an IPS (or IDS) use case.
    - Enables traffic inspection without the need for routing configurations. Allows traffic flow if the unit fails via Hardware Bypass.
- .Practical examples:
    - Set up layer 2 network inspection anywhere in a fast and easy way - without the need for layer 3.
    - Critical for networks that are fully isolated - no Internet access.
    - Transparent inline insertion for deep packet inspection for standalone firewall - existing production layer 2 architecture.

Basics: Supported Platforms, Licensing

Software & Hardware Versions

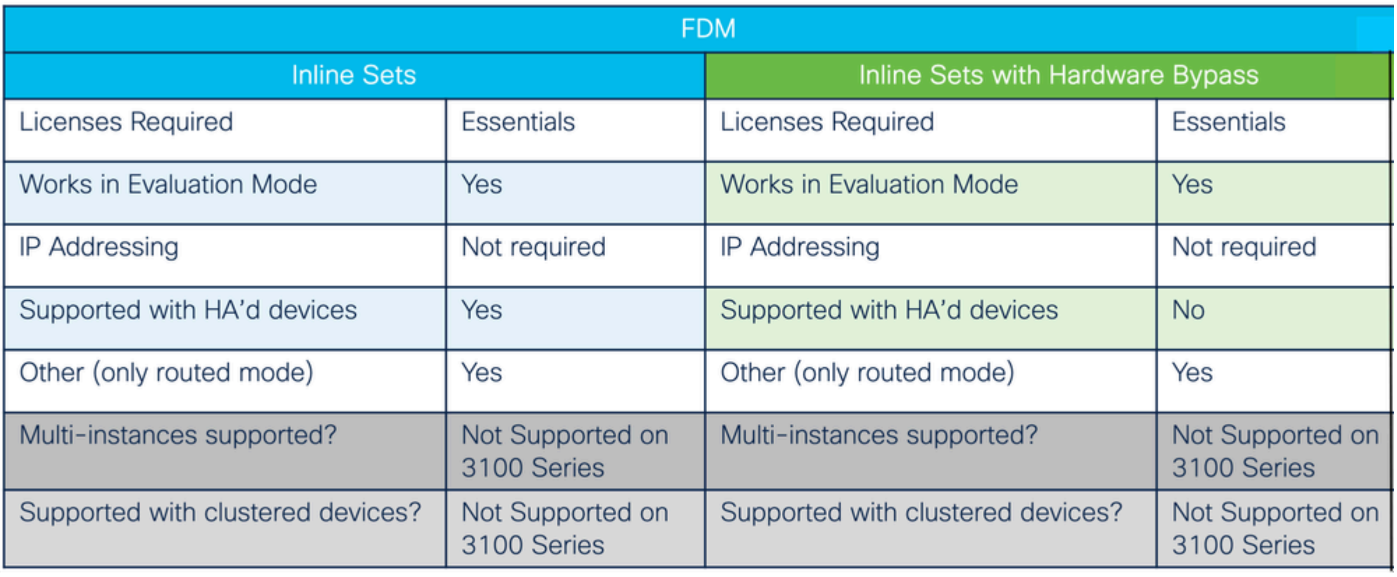


FDM		
	Inline Sets – before 7.7.0	Inline Sets with Hardware Bypass
FDM	7.4.1	7.7.0
REST API	7.4.1	7.7.0
Platforms	1000, 2100 (up to 7.4 only), and 3100 Series	3100 Series equipped with a network module: • 8 Ports: o FPR-X-NM-6X1SXF • 6 Ports: o FPR-X-NM-6X10SRF o FPR-X-NM-6X10LRF o FPR-X-NM-6X25SRF o FPR-X-NM-6X25LRF

Software and Hardware

Note: Information of 3100 Series and Hardware Bypass

Other Aspects of Support



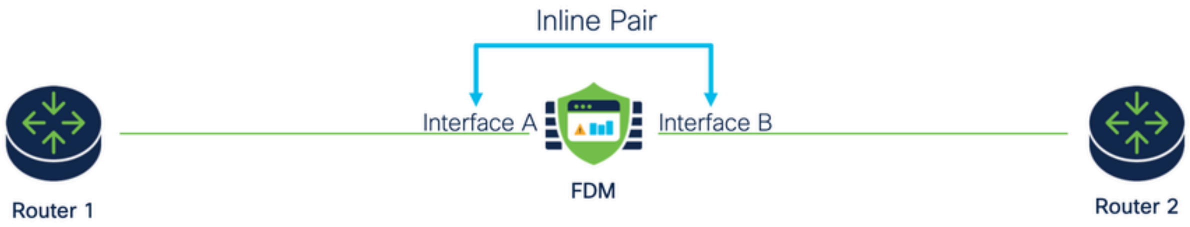
FDM			
Inline Sets		Inline Sets with Hardware Bypass	
Licenses Required	Essentials	Licenses Required	Essentials
Works in Evaluation Mode	Yes	Works in Evaluation Mode	Yes
IP Addressing	Not required	IP Addressing	Not required
Supported with HA'd devices	Yes	Supported with HA'd devices	No
Other (only routed mode)	Yes	Other (only routed mode)	Yes
Multi-instances supported?	Not Supported on 3100 Series	Multi-instances supported?	Not Supported on 3100 Series
Supported with clustered devices?	Not Supported on 3100 Series	Supported with clustered devices?	Not Supported on 3100 Series

Licensing and Compatibility

Feature Description and Walkthrough

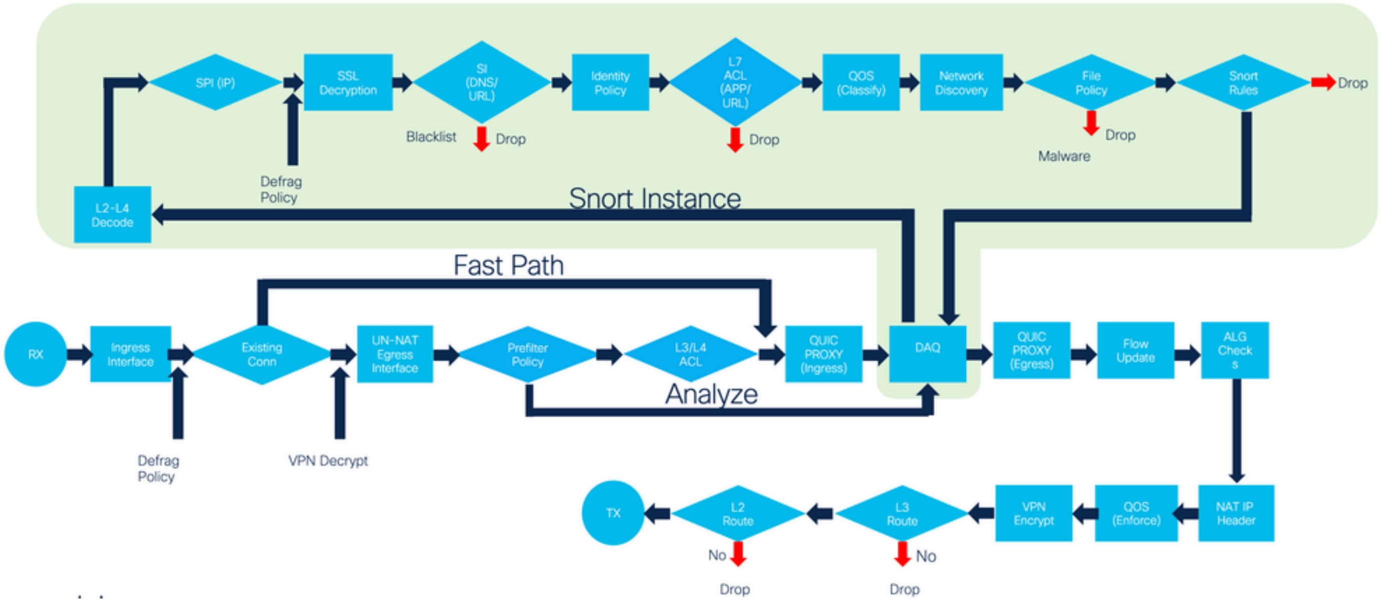
Functional Feature Description

- Inline Set Network Diagram



Inline Set Network Diagram

- Traffic flows from Router 1 to Router 2, through Interfaces A and B, using only a physical connection.
- FDM Inline Sets Packet Processing Flow Diagram:



Flow Diagram

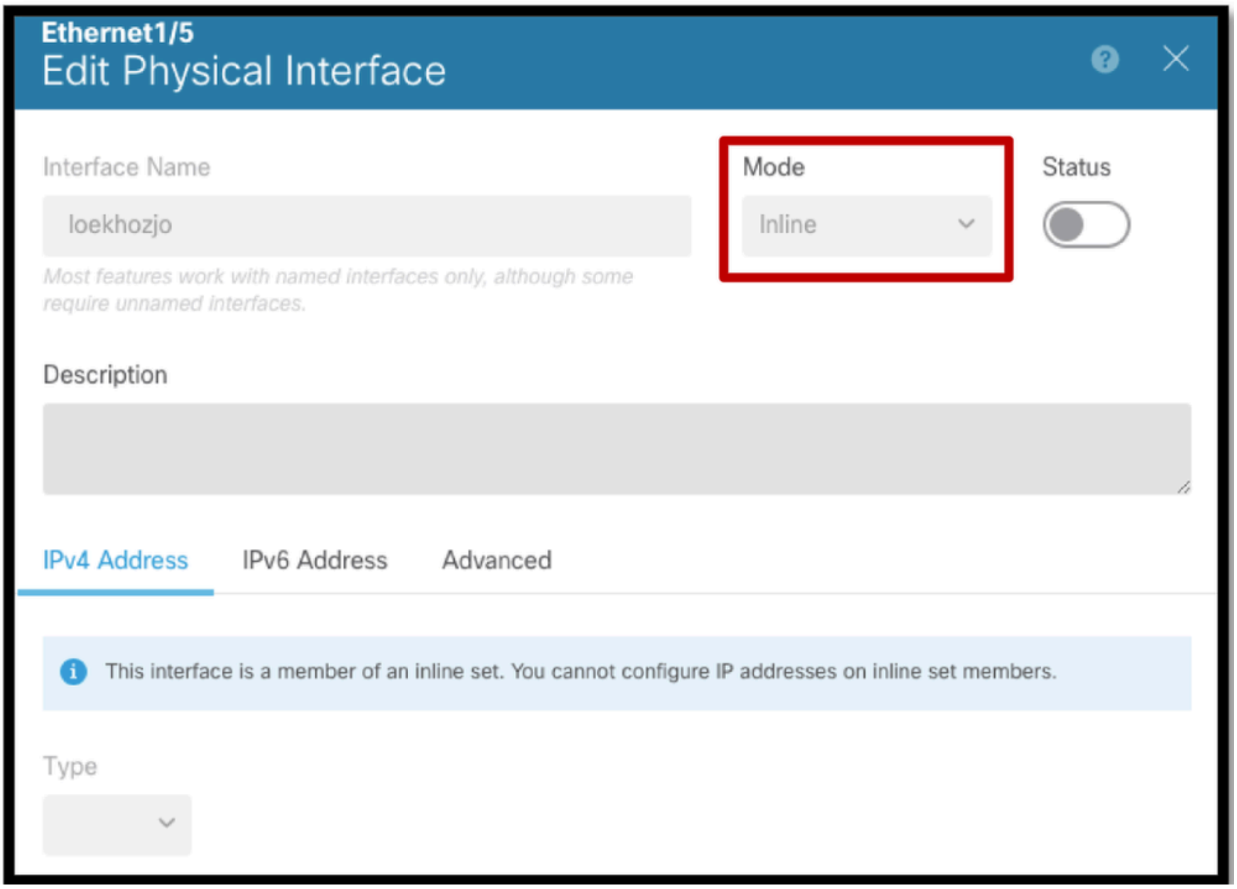
- Inline Sets:
  - Inline Sets are supported on physical interfaces and EtherChannels.

- Hardware Bypass:
  - Inline Sets with Hardware Bypass are supported on predetermined Physical Interface Pairs:
    Ethernet 1 & 2
    Ethernet 2 & 3
    Ethernet 4 & 5
    Ethernet 5 & 6

- Interface support:
  - Interfaces that are part of an Inline Pair:
    Must be named.
    Be free of any IP, DHCP, or PPPoE configurations.
    Must not be in Passive Mode.
    Must not be Management Interface.
    Must be used in only one Inline Pair at a time.

- Inline Mode Details
  - Inline Mode is available for Physical Interfaces, EtherChannels, and Security Zones.
  - Inline Mode is automatically set for Interfaces and EtherChannels when they are used in an Inline Pair.
  - Inline Mode prevents changes from being made on the involved Interfaces and EtherChannels until they are removed from the Inline Pair.
  - Interfaces that are in Inline Mode can be associated with Security Zones set to Inline Mode.

- Inline Mode GUI
  - The Edit Interface dialog reflects that the interface or EtherChannel is in Inline mode.
  - Changes are not permitted on interfaces when they are in Inline Mode. The Edit Physical Interface (or Edit EtherChannel) dialog is read-only.


Ethernet1/5

Edit Physical Interface

Interface Name: loekhozjo

Most features work with named interfaces only, although some require unnamed interfaces.

Mode: Inline

Status

Description

This interface is a member of an inline set. You cannot configure IP addresses on inline set members.

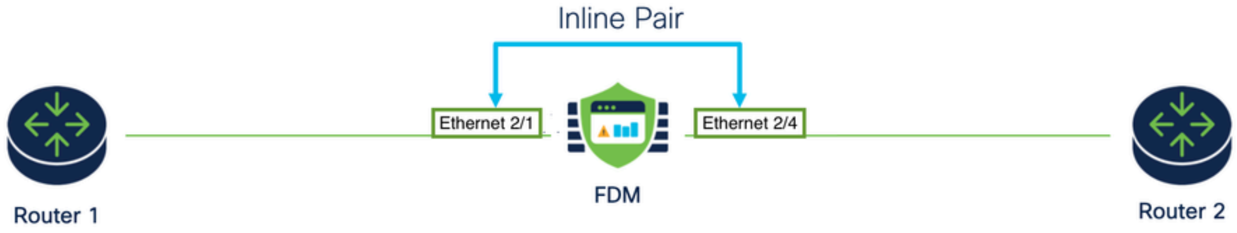
Type

Edit Interface in GUI

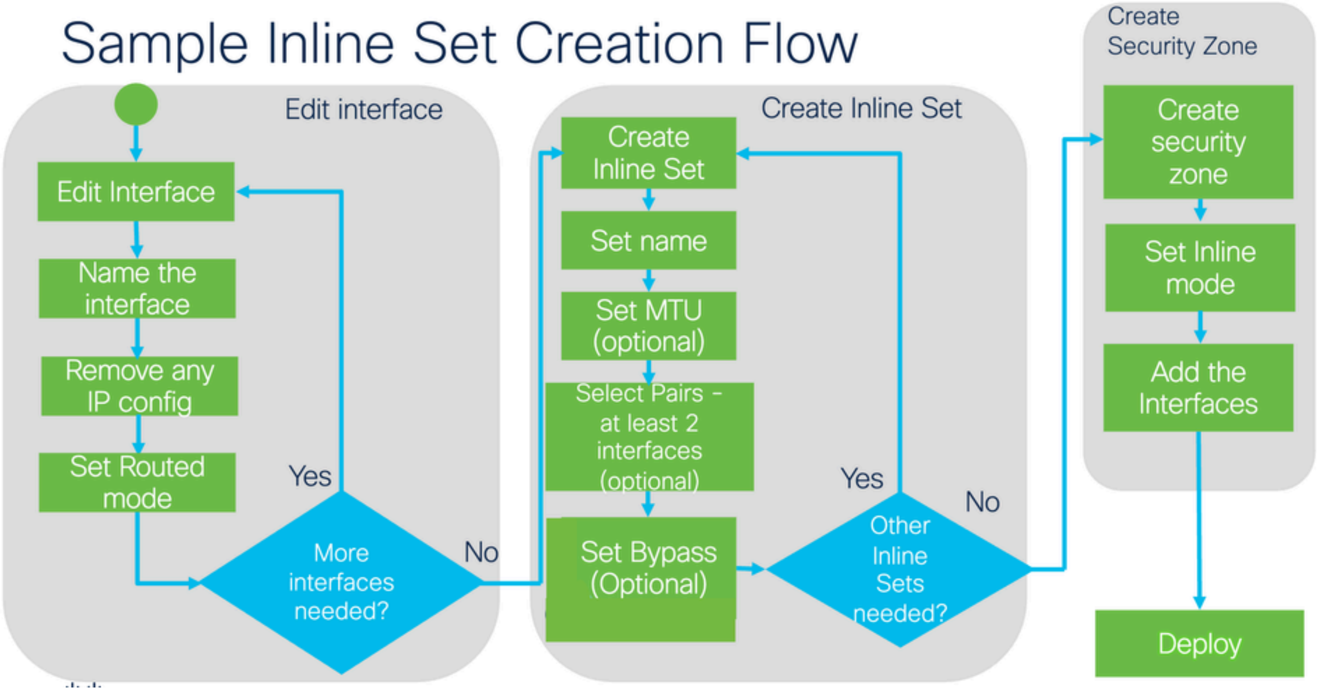
- Upgrade, Import/Export, Backup/Restore, Deploy
  - Upgrade Implications
    The user can upgrade FDM without any restrictions.
    When upgrading from an earlier version, existing Inline Set Objects are configured with their bypass field set to Disabled.
  - Import/Export Implications
    Inline Set objects are imported and exported.
  - Backup/Restore
    Inline Set objects are handled during Backup/Restore.
  - Deploy
    Objects deploy normally.
    Specific errors were implemented.

Configure

Network Diagram



Network Diagram



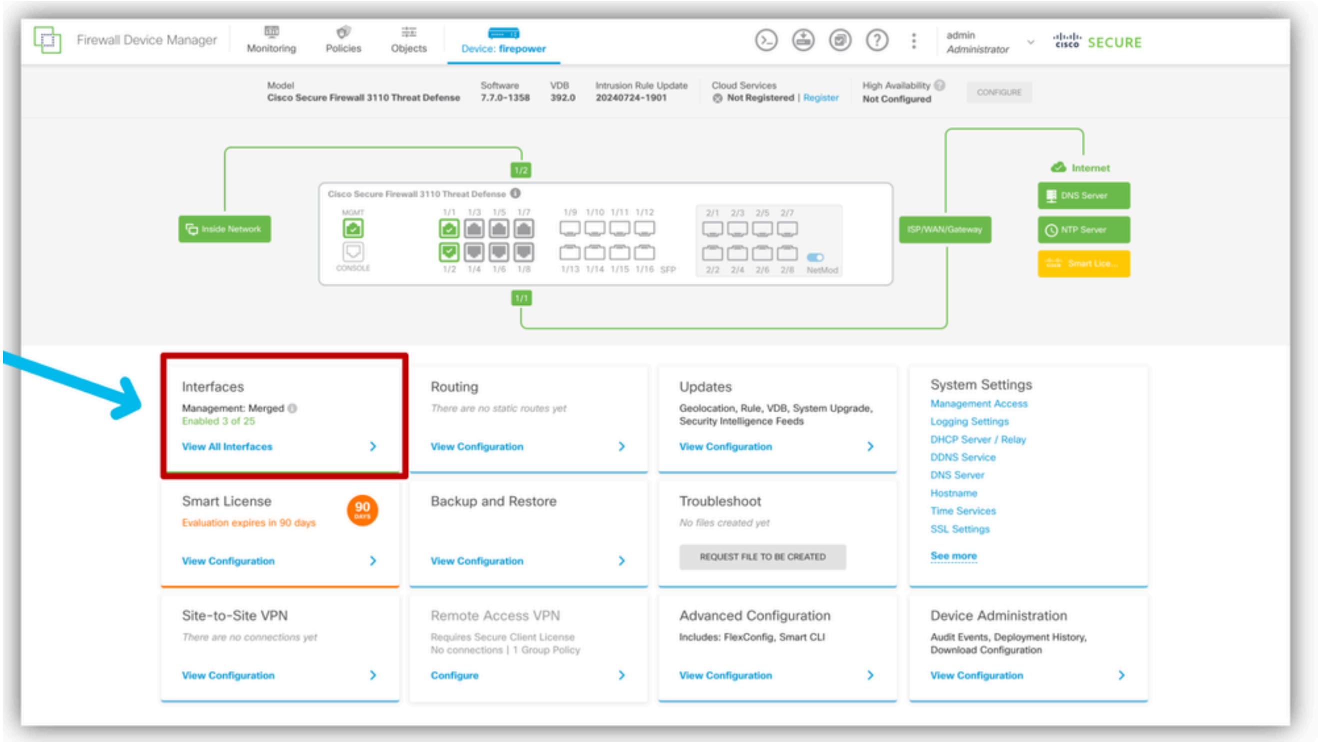
Inline Set Creation Flow

Configurations

This section describes the steps to configure Hardware Bypass on FDM

Step 1: Edit Interfaces.

- Log in to **FDM** and navigate to **Interface Management.**
- From the **FDM** dashboard, click the **Interfaces** card.



Model
Cisco Secure Firewall 3110 Threat Defense

Software
7.7.0-1358

VDB
392.0

Intrusion Rule Update
20240724-1901

Cloud Services
Not Registered | Register

High Availability
Not Configured

CONFIGURE

Interfaces
Management: Merged
Enabled 3 of 25
View All Interfaces

Routing
There are no static routes yet
View Configuration

Updates
Geolocation, Rule, VDB, System Upgrade, Security Intelligence Feeds
View Configuration

System Settings
Management Access
Logging Settings
DHCP Server / Relay
DDNS Service
DNS Server
Hostname
Time Services
SSL Settings
See more

Smart License
Evaluation expires in 90 days
90 DAYS
View Configuration

Backup and Restore
View Configuration

Troubleshoot
No files created yet
REQUEST FILE TO BE CREATED

Site-to-Site VPN
There are no connections yet
View Configuration

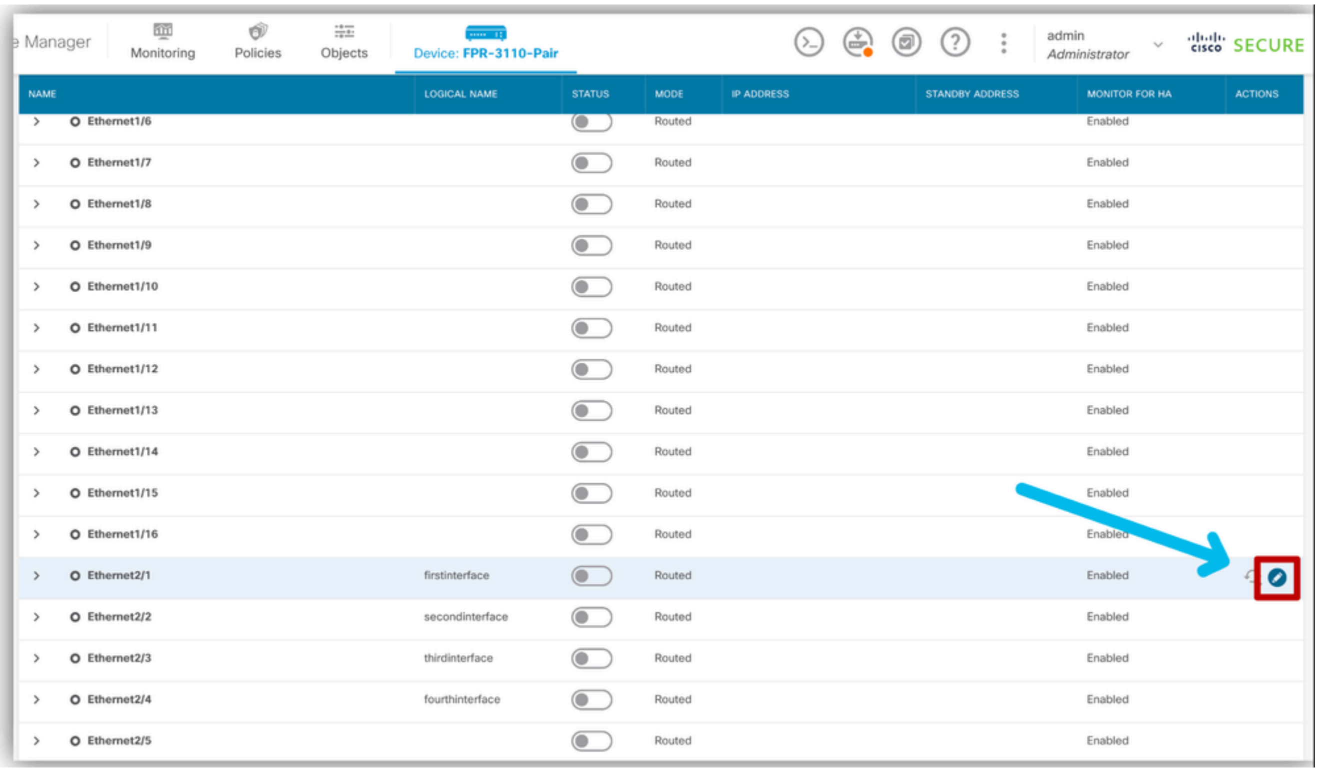
Remote Access VPN
Requires Secure Client License
No connections | 1 Group Policy
Configure

Advanced Configuration
Includes: FlexConfig, Smart CLI
View Configuration

Device Administration
Audit Events, Deployment History, Download Configuration
View Configuration

Select Interface

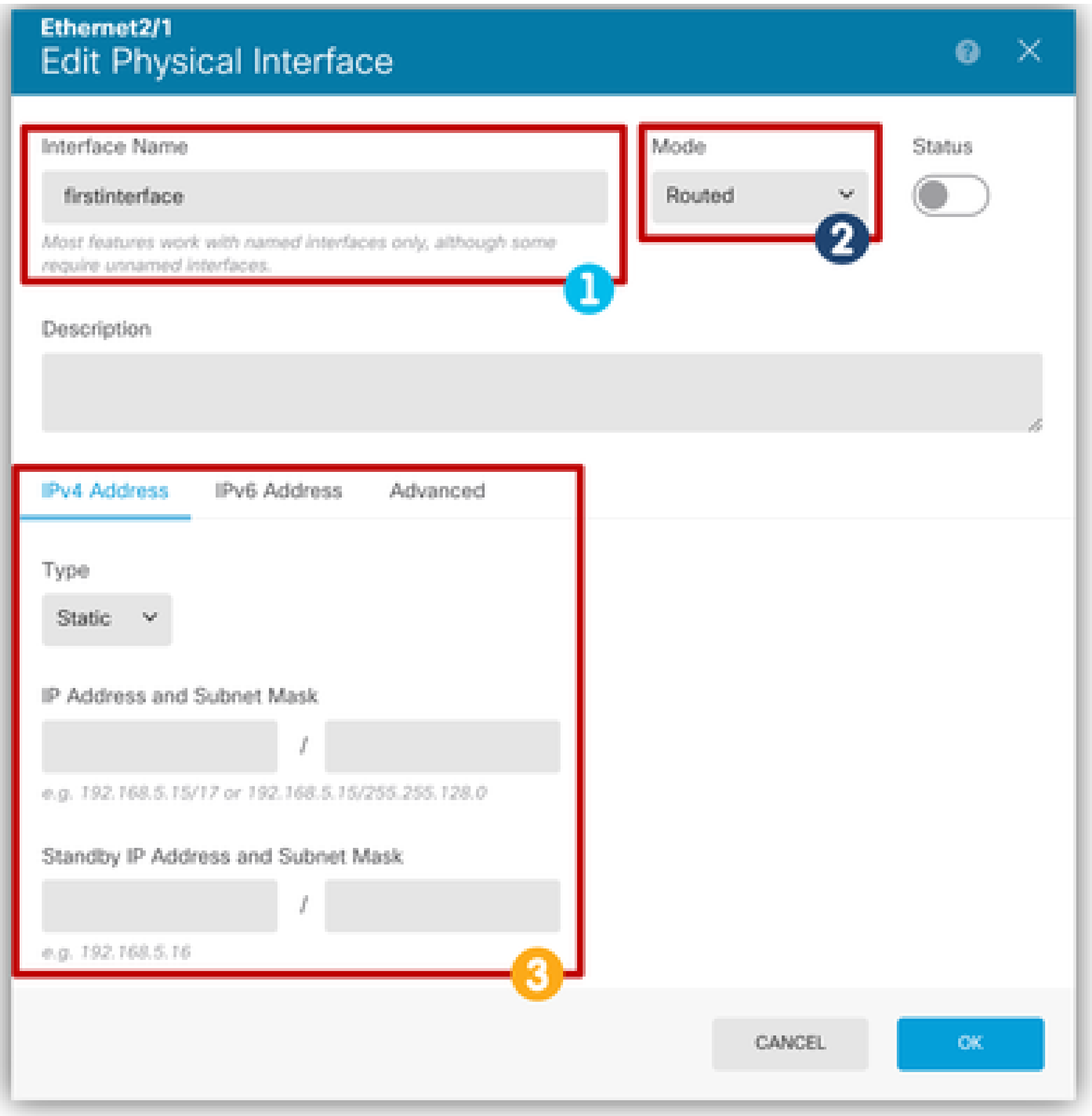
- **Edit** the interfaces that are used in the Inline Set.
- To **Edit** interfaces, click the **edit (pencil)** icon for the interface.



NAME	LOGICAL NAME	STATUS	MODE	IP ADDRESS	STANDBY ADDRESS	MONITOR FOR HA	ACTIONS
Ethernet1/6			Routed			Enabled	
Ethernet1/7			Routed			Enabled	
Ethernet1/8			Routed			Enabled	
Ethernet1/9			Routed			Enabled	
Ethernet1/10			Routed			Enabled	
Ethernet1/11			Routed			Enabled	
Ethernet1/12			Routed			Enabled	
Ethernet1/13			Routed			Enabled	
Ethernet1/14			Routed			Enabled	
Ethernet1/15			Routed			Enabled	
Ethernet1/16			Routed			Enabled	
Ethernet2/1	firstinterface		Routed			Enabled	
Ethernet2/2	secondinterface		Routed			Enabled	
Ethernet2/3	thirdinterface		Routed			Enabled	
Ethernet2/4	fourthinterface		Routed			Enabled	
Ethernet2/5			Routed			Enabled	

Edit Interface

- Edit physical interface:
  1. Name the **Interface**.
  2. Select **Routed** Mode.
  3. Remove any **IP configuration**.



Ethernet2/1

Edit Physical Interface

Interface Name

firstinterface

Most features work with named interfaces only, although some require unnamed interfaces.

Mode

Routed

Status

Description

IPv4 Address | IPv6 Address | Advanced

Type

Static

IP Address and Subnet Mask

e.g. 192.168.5.15/17 or 192.168.5.15/255.255.128.0

Standby IP Address and Subnet Mask

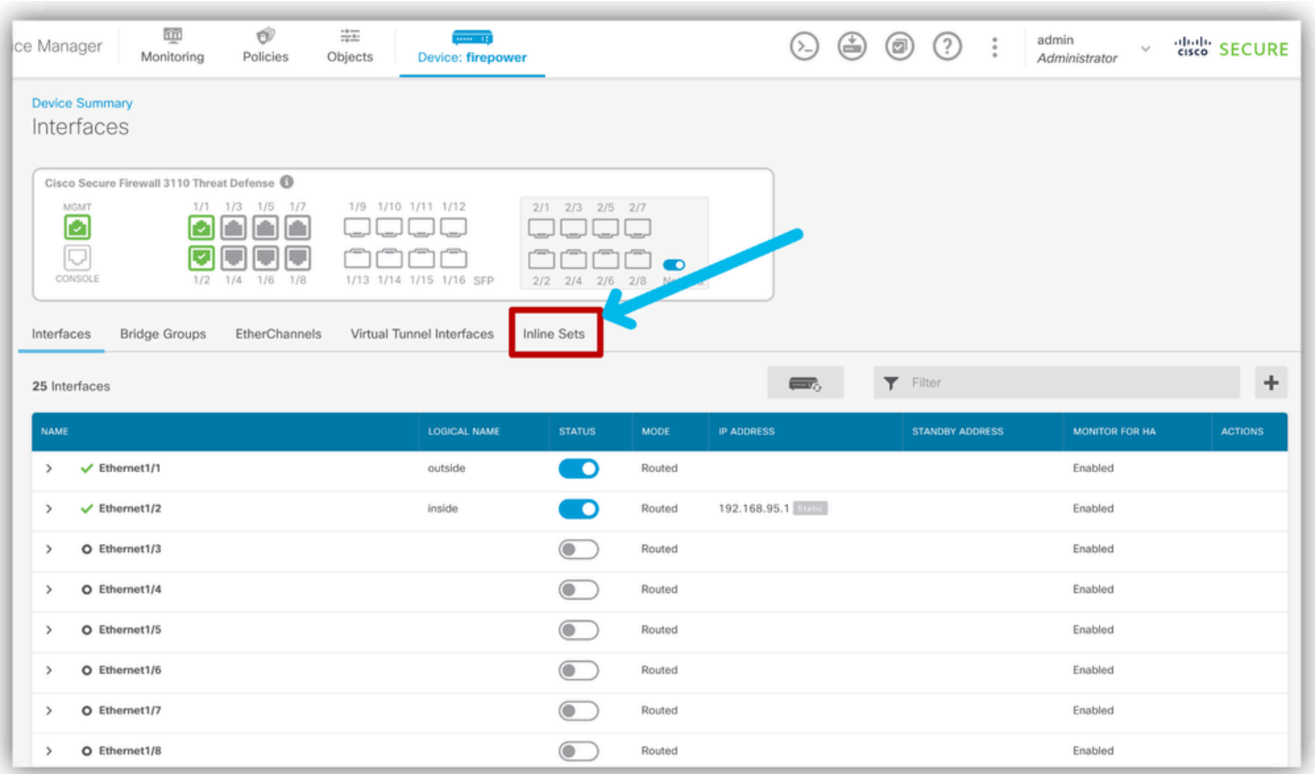
e.g. 192.168.5.16

Configure Parameters

Note: The mode is automatically changed to Inline after the interface is added in an Inline Pair.

Step 2: Create an Inline Set.

- Navigate to **Device > Interfaces > Inline sets** tab.

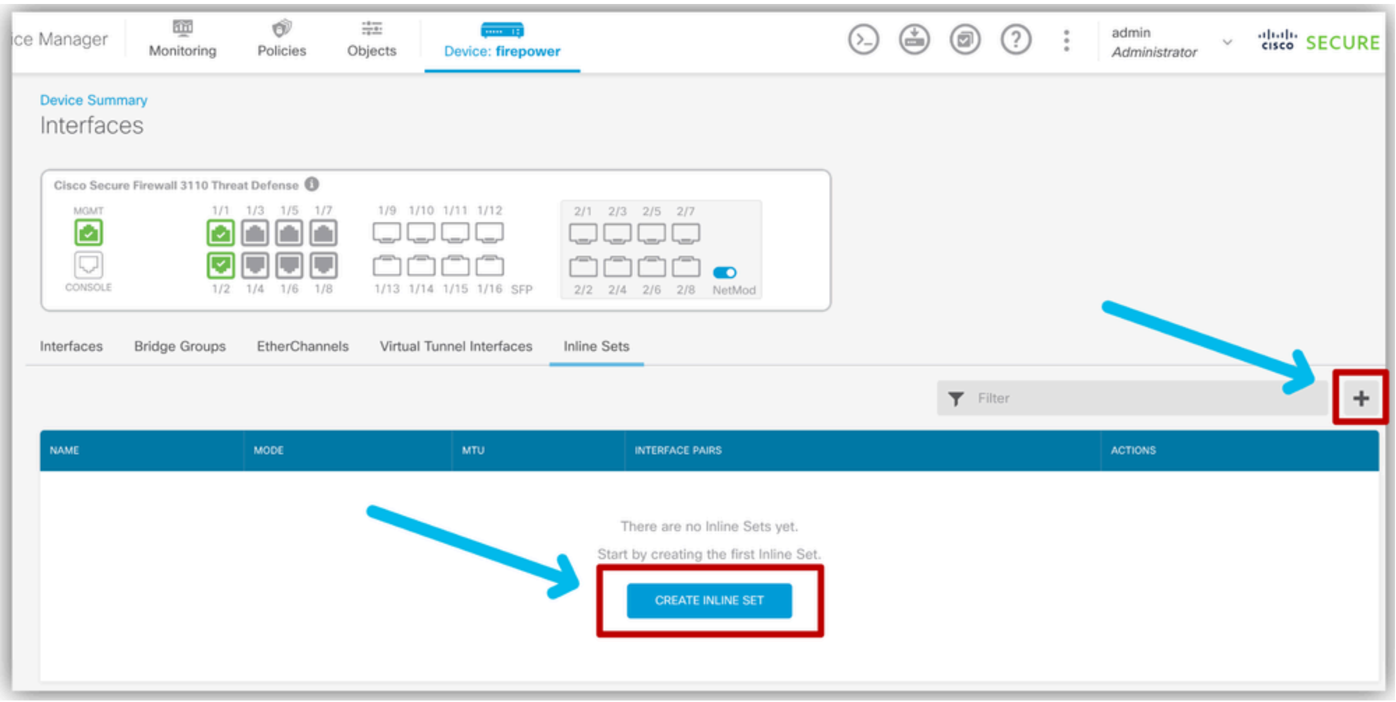


25 Interfaces

NAME	LOGICAL NAME	STATUS	MODE	IP ADDRESS	STANDBY ADDRESS	MONITOR FOR HA	ACTIONS
✓ Ethernet1/1	outside	●	Routed			Enabled	
✓ Ethernet1/2	inside	●	Routed	192.168.95.1 Static		Enabled	
○ Ethernet1/3		○	Routed			Enabled	
○ Ethernet1/4		○	Routed			Enabled	
○ Ethernet1/5		○	Routed			Enabled	
○ Ethernet1/6		○	Routed			Enabled	
○ Ethernet1/7		○	Routed			Enabled	
○ Ethernet1/8		○	Routed			Enabled	

Navigate To Inline Sets Tab

- Add a new **Inline Set**.
- Click + icon or **Create Inline Set** Button.



NAME	MODE	MTU	INTERFACE PAIRS	ACTIONS

There are no Inline Sets yet.

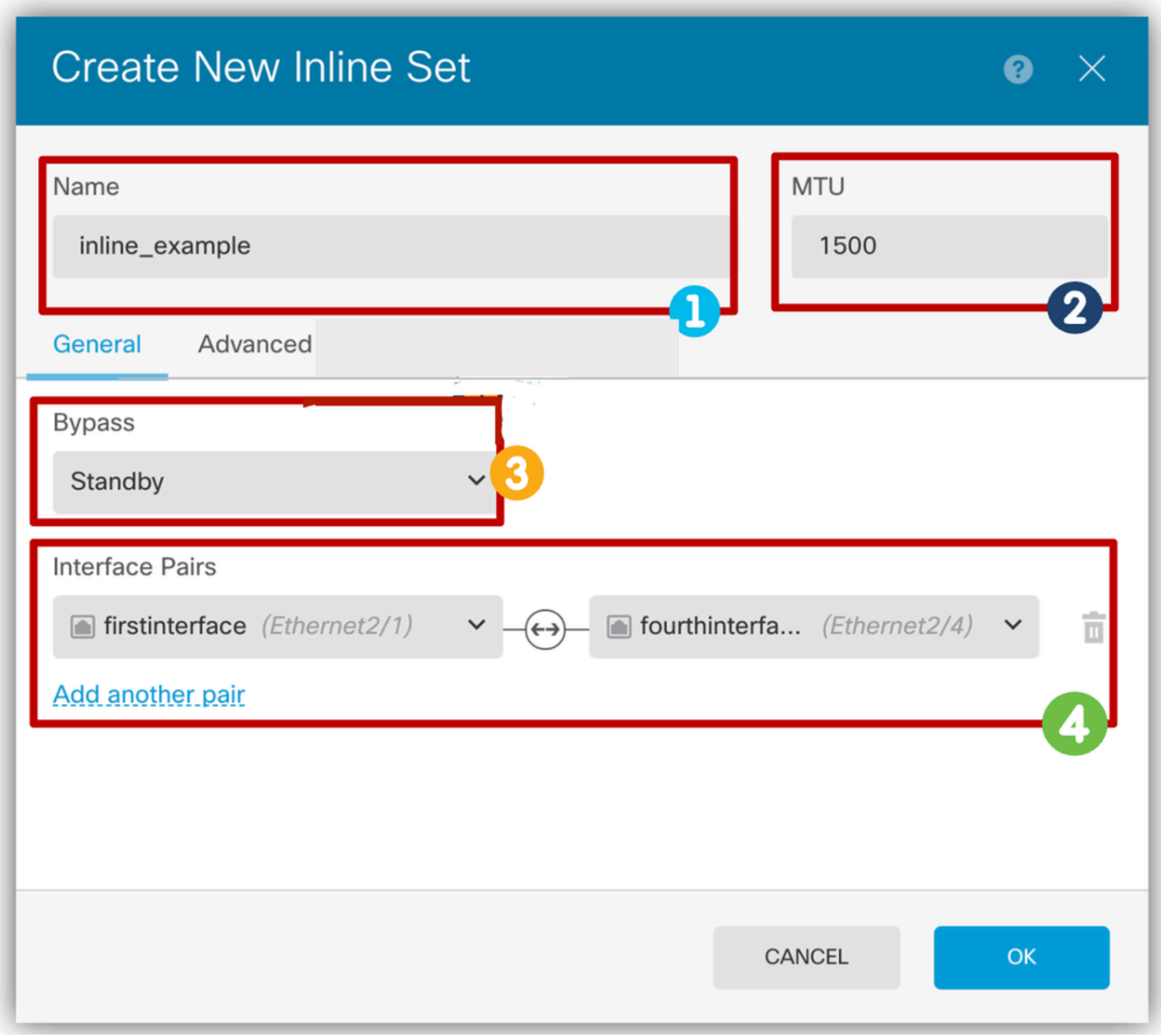
Start by creating the first Inline Set.

Create Inline Set

.

- Configure basic settings.
    1. Set a **Name**.

2. Set desired **MTU** (optional). The default is **1500**, which is the minimum supported MTU.
3. Select hardware **Bypass** (Details available in the next section). A new dropdown menu was added for **Bypass**.
4. In the **Interface Pairs** section select **interfaces**.
5. Named intefaces are available forselection. If more pairs are required, click the **Add another pair** link.



Create New Inline Set

Name: inline_example

MTU: 1500

Bypass: Standby

Interface Pairs: firstinterface (Ethernet2/1) ↔ fourthinterfa... (Ethernet2/4)

Configure Settings

Hardware Bypass

Capabilities and Limitations

- 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.
- Hardware Bypass ports are supported only for Inline Sets.
- Hardware Bypass is NOT supported in High Availability mode.
- Hardware Bypass modes:
  - DISABLED - Disables bypass on supported interfaces. Default mode for unsupported

interfaces.

  - STANDBY - In the standby state, the interfaces remain in normal operation until there is a trigger event.
  - BYPASS FORCE - Manually forces the interface pair to bypass inspection.

Note: Information on FTD Interface types and Hardware Bypass

Snort Fail Open vs Hardware Bypass

- Hardware Bypass functionality allows traffic to flow during a hardware failure, including a complete power outage, and certain limited software failures.
- A software failure that triggers Snort Fail Open does not trigger aHardware Bypass.

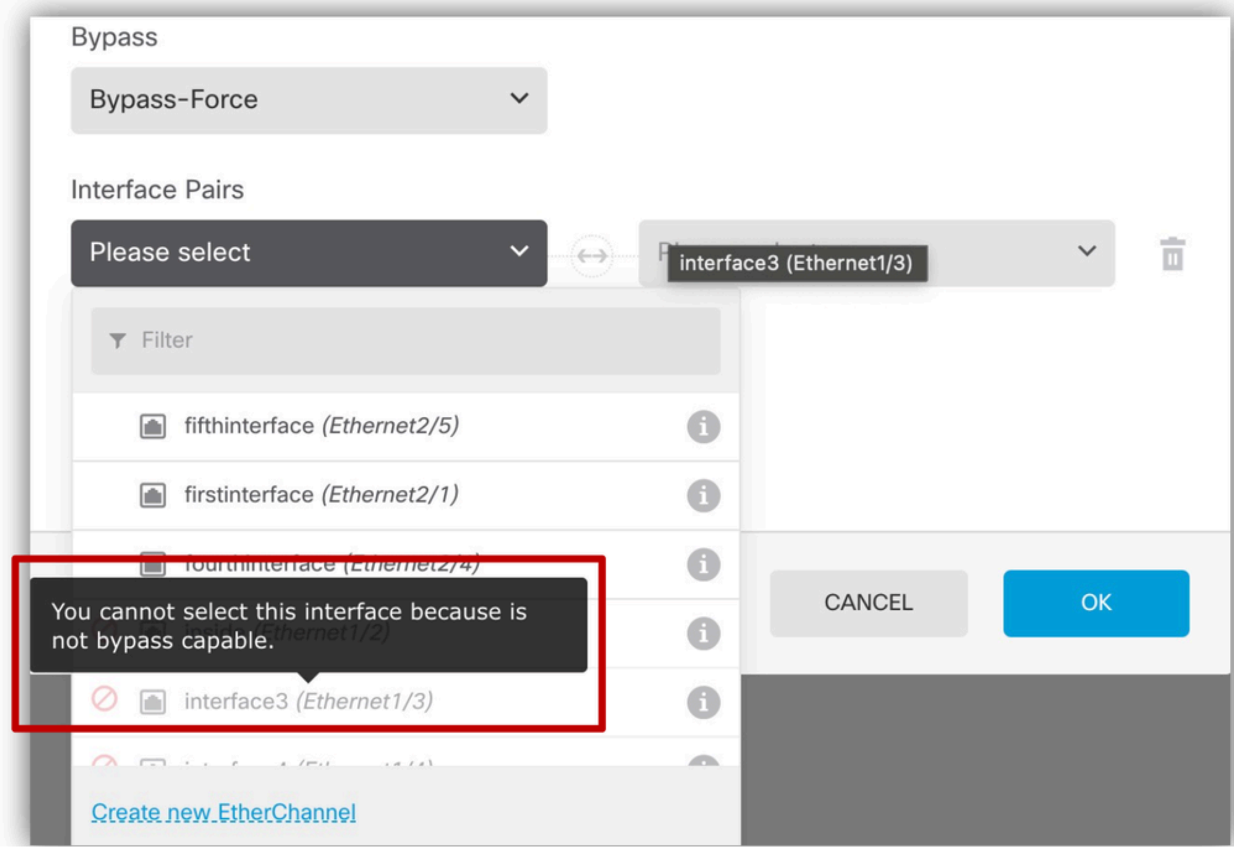
Hardware BypassTriggers

Hardware Bypass can be triggered in the folowing scenarios:

- Application crash
- Application reboot
- Device crash
- Device reboot or upgrade
- Device power loss
- Manual trigger

To see which interfaces support Hardware Bypass:

- From the FDM GUI, if Bypass is selected:
  - Interfaces that support it are selectable.
  - Interfaces that do not support are grayed out.
  - For this example, Ethernet1/3 is grayed out in this figure:



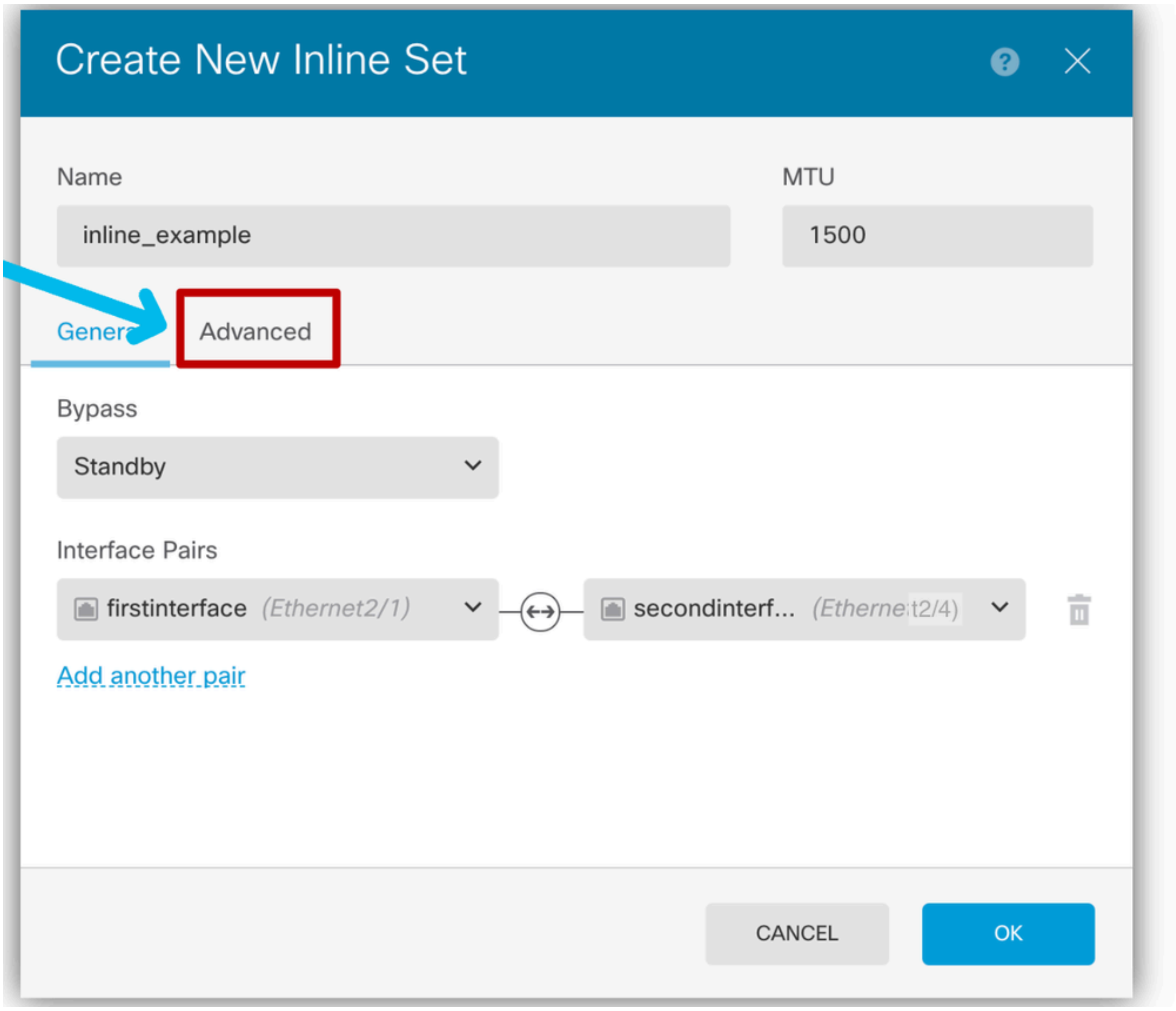
Bypass: Bypass-Force

You cannot select this interface because is not bypass capable.

Verify Hardware Bypass Support

Step 3: Configure inline sets **Advanced** Setting.

- Navigate to **Device** > **Interfaces** > **Inline sets** tab or **edit** an already created inline set.
- Navigate to the **Advanced** Tab.
  - The Advanced tab allows you to configure the setting of options for Inline Sets.
  - Click the **Advanced** tab.



Create New Inline Set

Name

inline_example

MTU

1500

Bypass

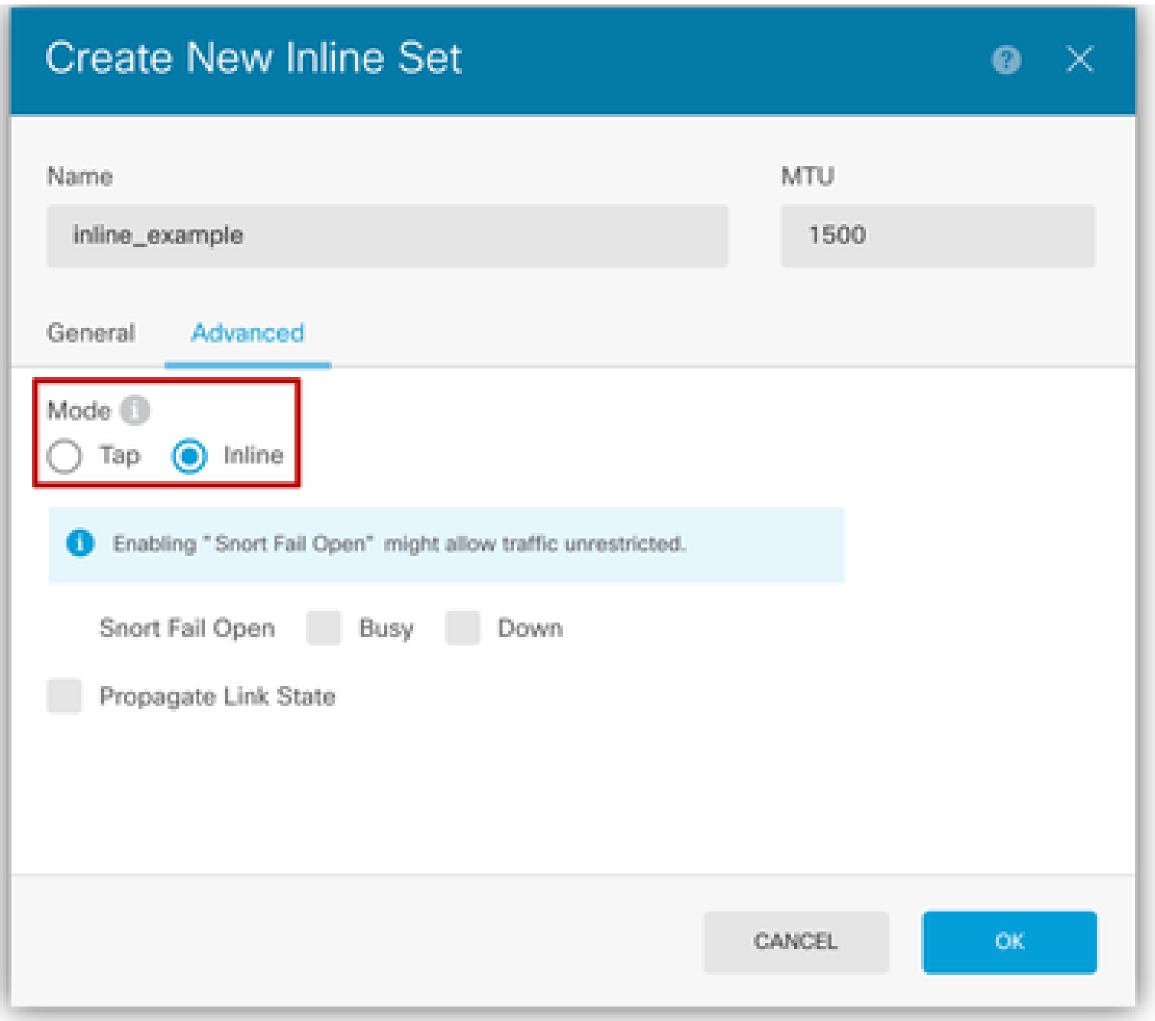
Standby

Interface Pairs

firstinterface (Ethernet2/1) — secondinterf... (Ethernet2/4)

Configure Inline Set

- **Mode**
  - Tap: Sets to inline tap mode,if Tap Mode is enabled, Snort Fail Open is disabled.
  - Inline



Create New Inline Set

Name: inline_example

MTU: 1500

Mode

○ Tap ● Inline

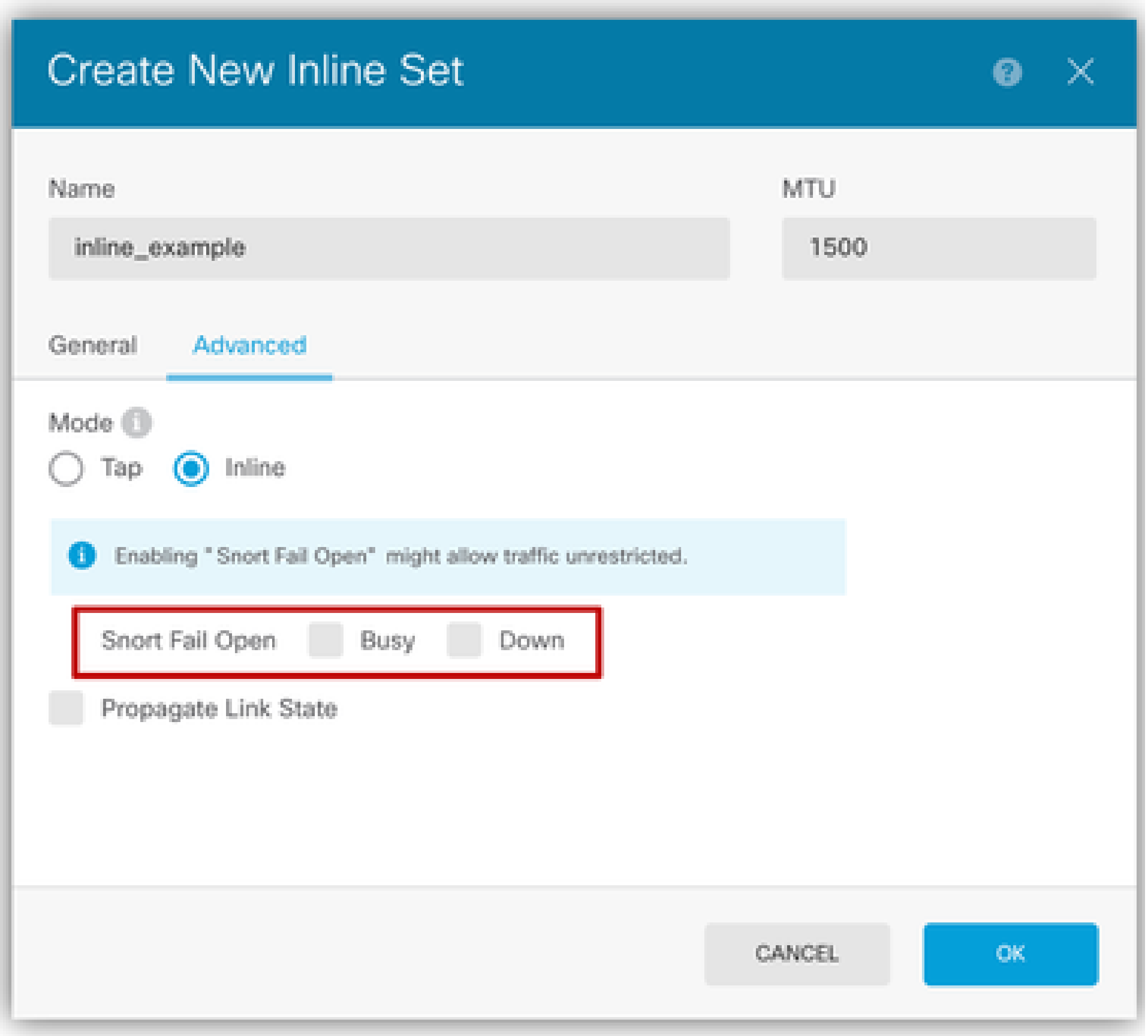
Enabling "Snort Fail Open" might allow traffic unrestricted.

Snort Fail Open □ Busy □ Down

□ Propagate Link State

Select Mode

- Snort Fail Open Settings.
  - Pick desired **Snort Fail Open** settings.
  - **None**, **one** or **both**. **Busy** and **Down** options can be set.
  - Snort Fail Open allows new and existing traffic to pass without inspection (enabled) or drop (disabled) when the Snort process is busy or down.



Create New Inline Set

Name: inline_example

MTU: 1500

General | Advanced

Mode

○ Tap ● Inline

Enabling " Snort Fail Open" might allow traffic unrestricted.

Snort Fail Open □ Busy □ Down

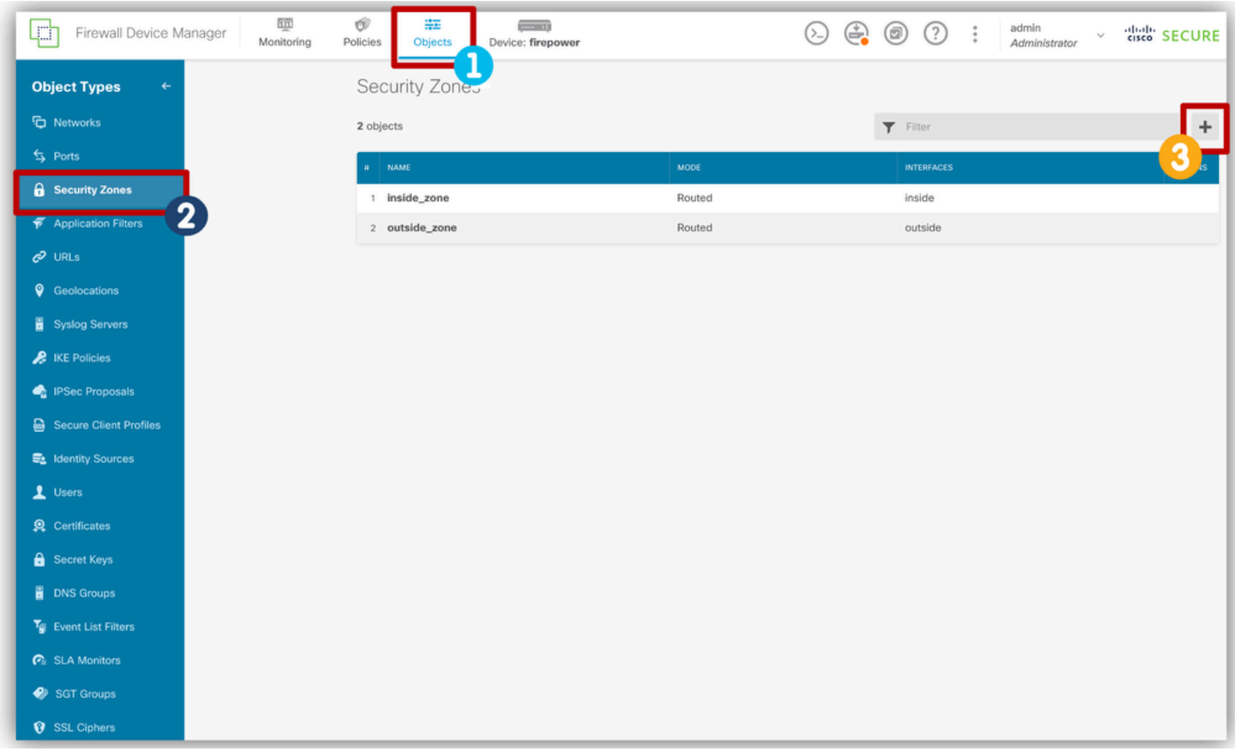
□ Propagate Link State

Snort Fail Open and Propagate Link State

- Propagate Link State.
  - Propagate Link State automatically brings down the second interface in the Inline pair when one of the iinterfaces goes down. When the downed interface comes back up, the second interface also automatically comes back up.
- Click **OK** to create the inline set.

Step 4: Apply to a Security Zone (optional).

1. From the top navigation bar, navigate to **Objects**.
2. Pick **Security Zones** from the left navigation:
   - Click **+** to add **security zone**.

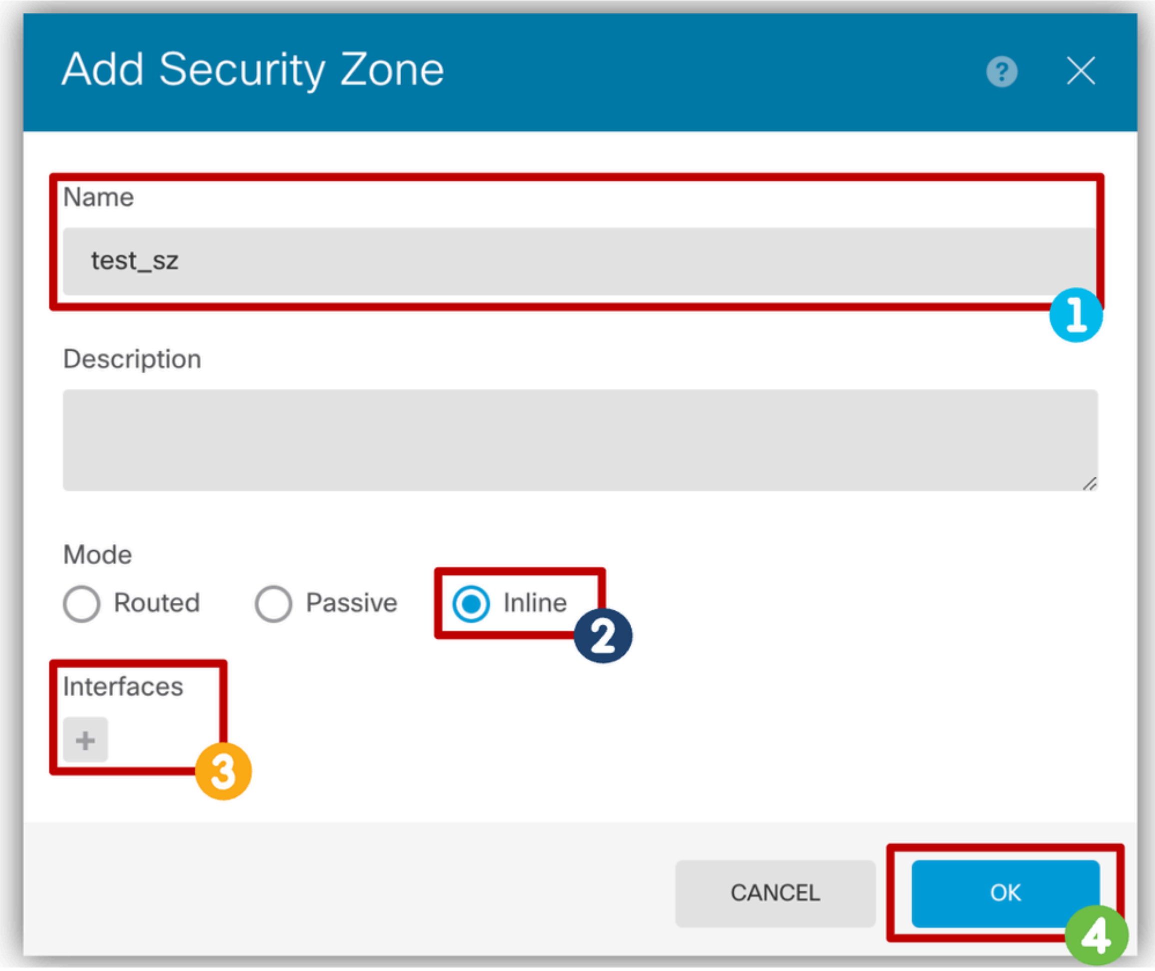


#	NAME	MODE	INTERFACES
1	inside_zone	Routed	inside
2	outside_zone	Routed	outside

Add a Security Zone

Configure the security zone(optional)

1. Name the **Security Zone**.
2. Select **Inline** Mode.
   Security Zones and Interfaces need to have the same mode.
3. Select **Interfaces** that are part of the inline set.
4. Click **OK**.



Add Security Zone

Name

test_sz

Description

Mode

○ Routed ○ Passive ● Inline

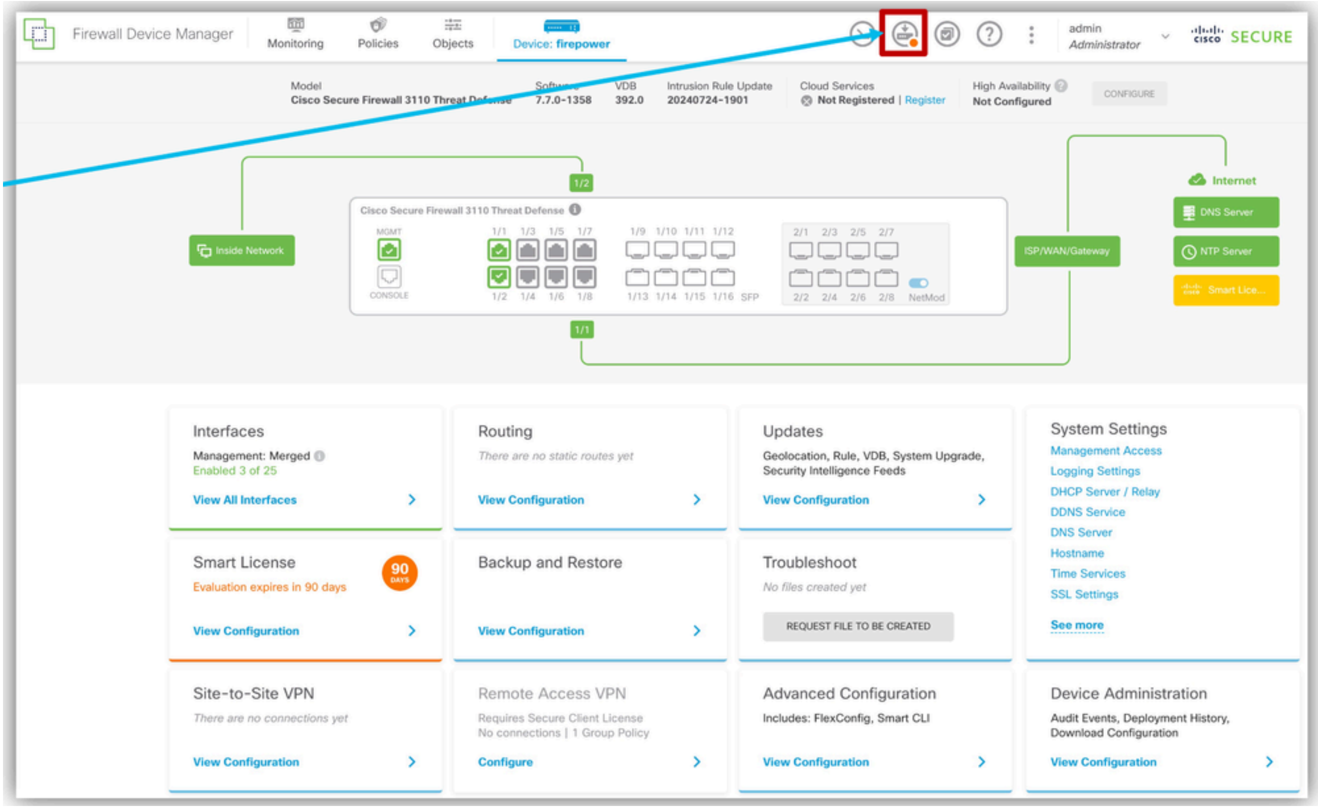
Interfaces

Configure Security Zone

Note: For interfaces, the mode automatically changed to Inline after the interface is added in an Inline Pair.

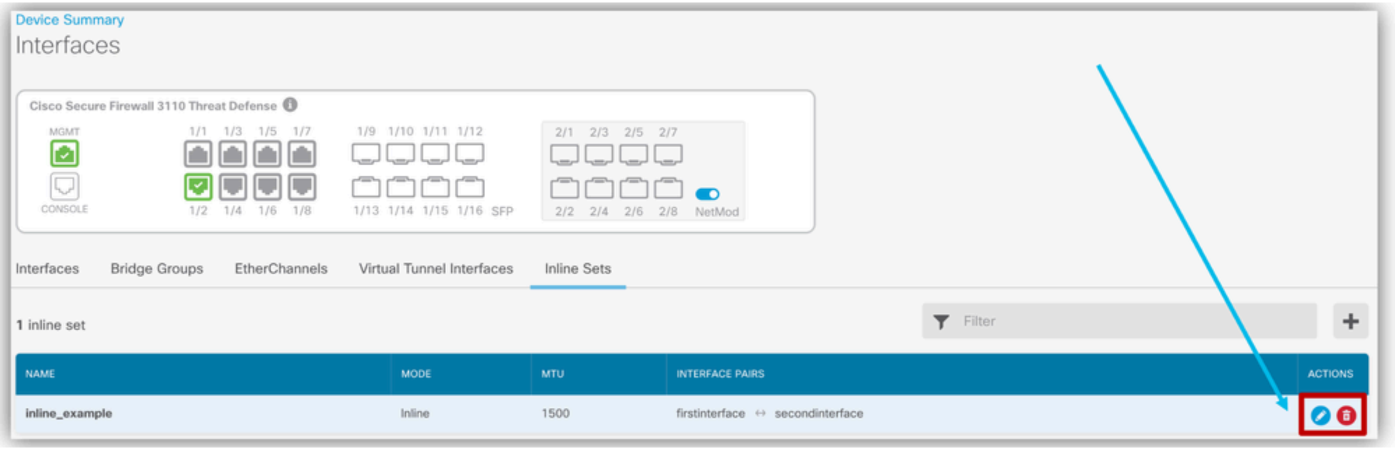
Step 4: Deploy

- Navigate to **Deployment** tab and **deploy**.



Deploy Changes

- Edit and Delete inline sets.
  - Navigate to **Device** > **Interfaces** > **Inline sets** tab.
  - Edit and Delete buttons are available for inline sets.



1 inline set

NAME	MODE	MTU	INTERFACE PAIRS	ACTIONS
inline_example	Inline	1500	firstinterface ↔ secondinterface	

Edit and Delete Inline Sets

FDM Device REST APIs

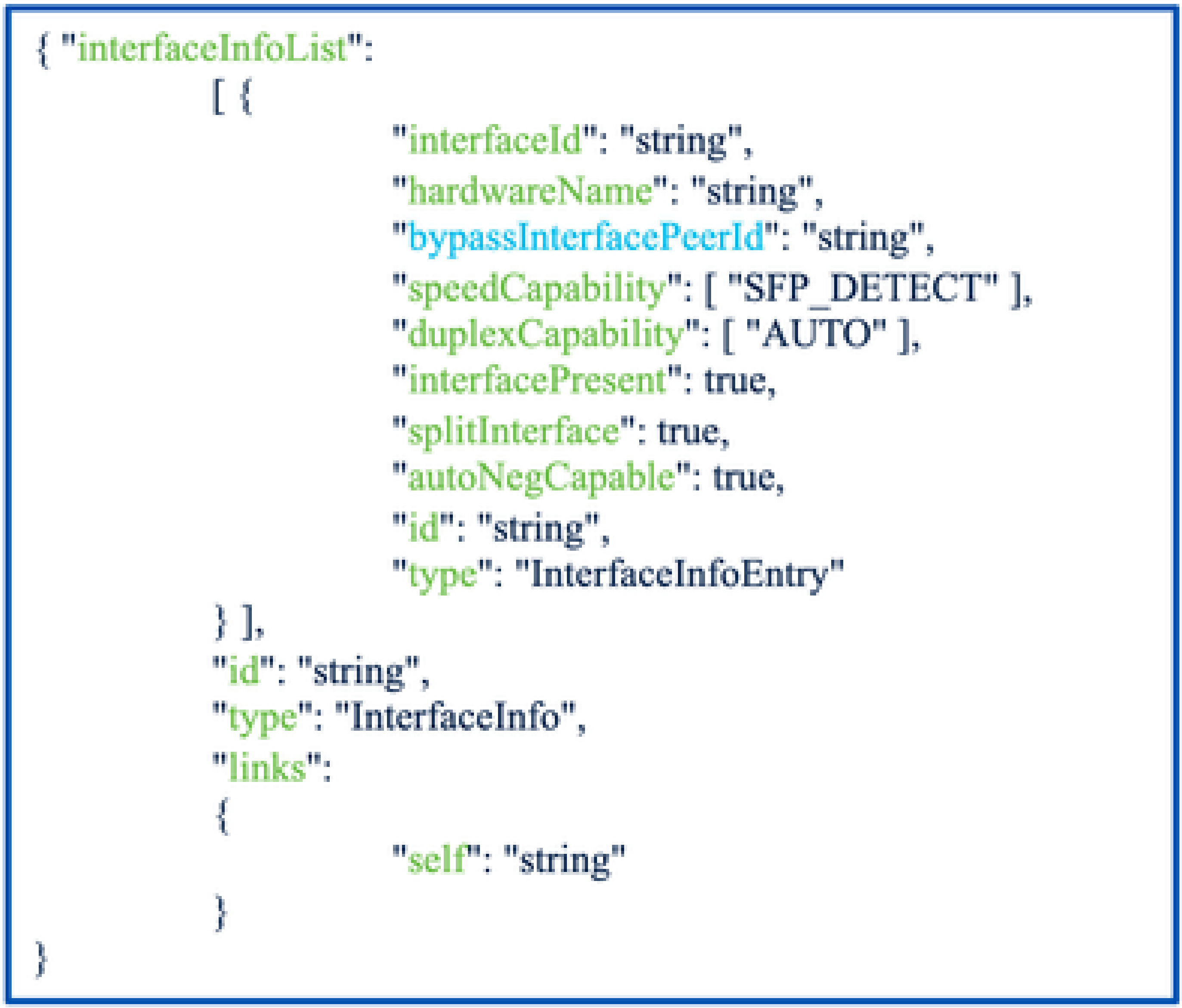
REST API Endpoints

- **GET : /devices/default/inlinesets**
  Fetch a list of all existing inline-sets.
- **GET :/devices/default/inlinesets/{objID}**
  Fetch a specific inline-set object by its ID.

- **POST : /devices/default/inlinesets**
  Create a new inline-set.
- **PUT : /devices/default/inlinesets/{objID}**
  Update existing inline-set object by its ID.
- **DELETE :/devices/default/inlinesets/{objID}**
  Delete existing inline-set object by its ID.
- **GET :/operational/interfaceinfo/{objID}**
  Fetch a list of all InterfaceInfoentities.
- To support Hardware Bypass, a new field was added to the InterfaceInfo API.

Interface Info REST API Models

- A new field bypassInterfacePeerIdwas added to aid Hardware Bypass integration.
- This field represents the ID of the Hardware Bypass Interface Pair for the current Interface.
- Values:
  - Null - interface does not support bypass.
  - ID - interface supports bypass.

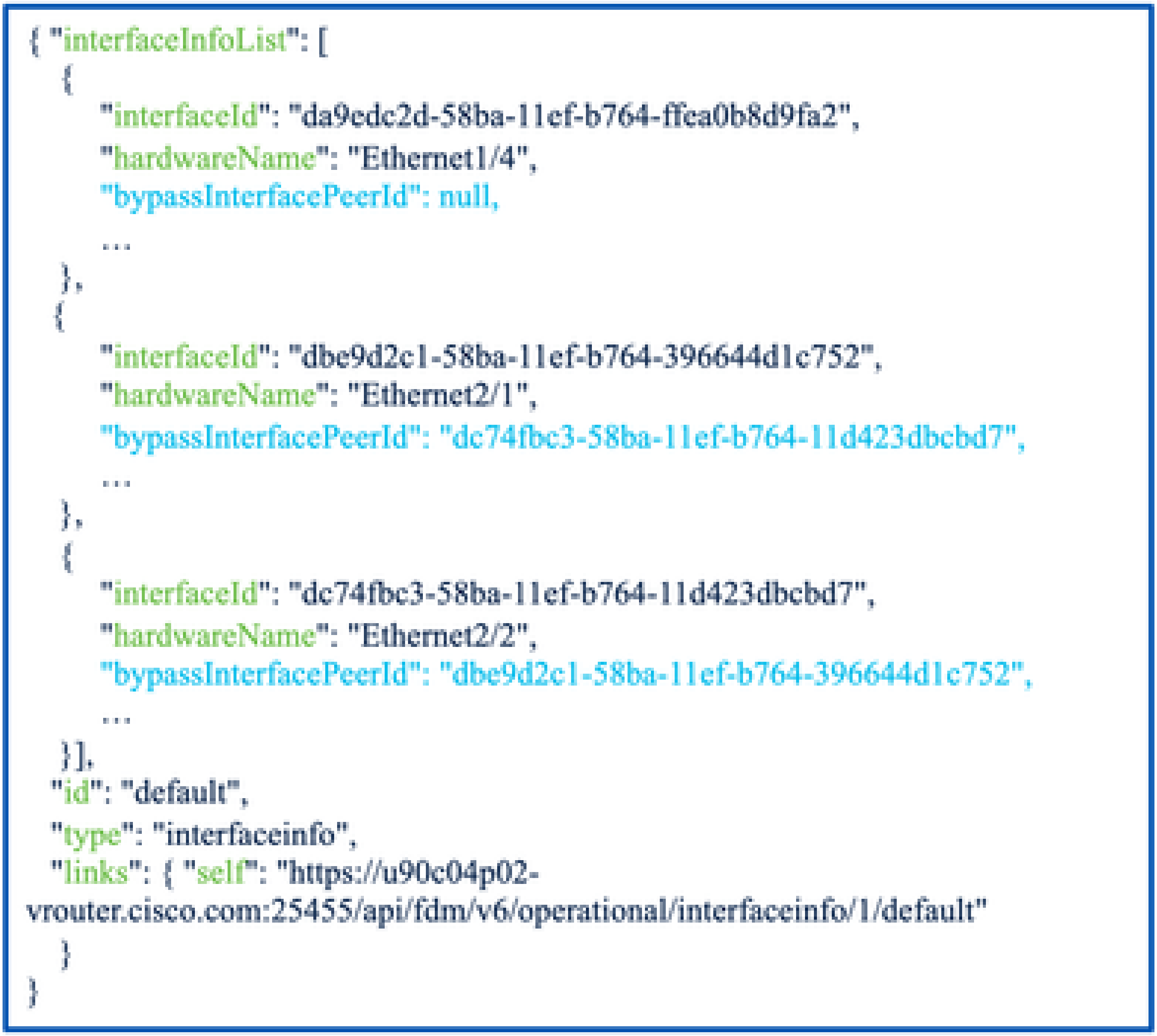


```
{ "interfaceInfoList":
            [ {
                        "interfaceId": "string",
                        "hardwareName": "string",
                        "bypassInterfacePeerId": "string",
                        "speedCapability": [ "SFP_DETECT" ],
                        "duplexCapability": [ "AUTO" ],
                        "interfacePresent": true,
                        "splitInterface": true,
                        "autoNegCapable": true,
                        "id": "string",
                        "type": "InterfaceInfoEntry"
            } ],
            "id": "string",
            "type": "InterfaceInfo",
            "links":
            {
                        "self": "string"
            }
}
```

Interface Info REST API

Interface Info REST API Example

- Interface Info REST API example.
  - Interface without Hardware Bypass support (Ethernet 1/4).
  - Interface Pair with Hardware Bypass Support (Ethernet2/1 and Ethernet 2/2).



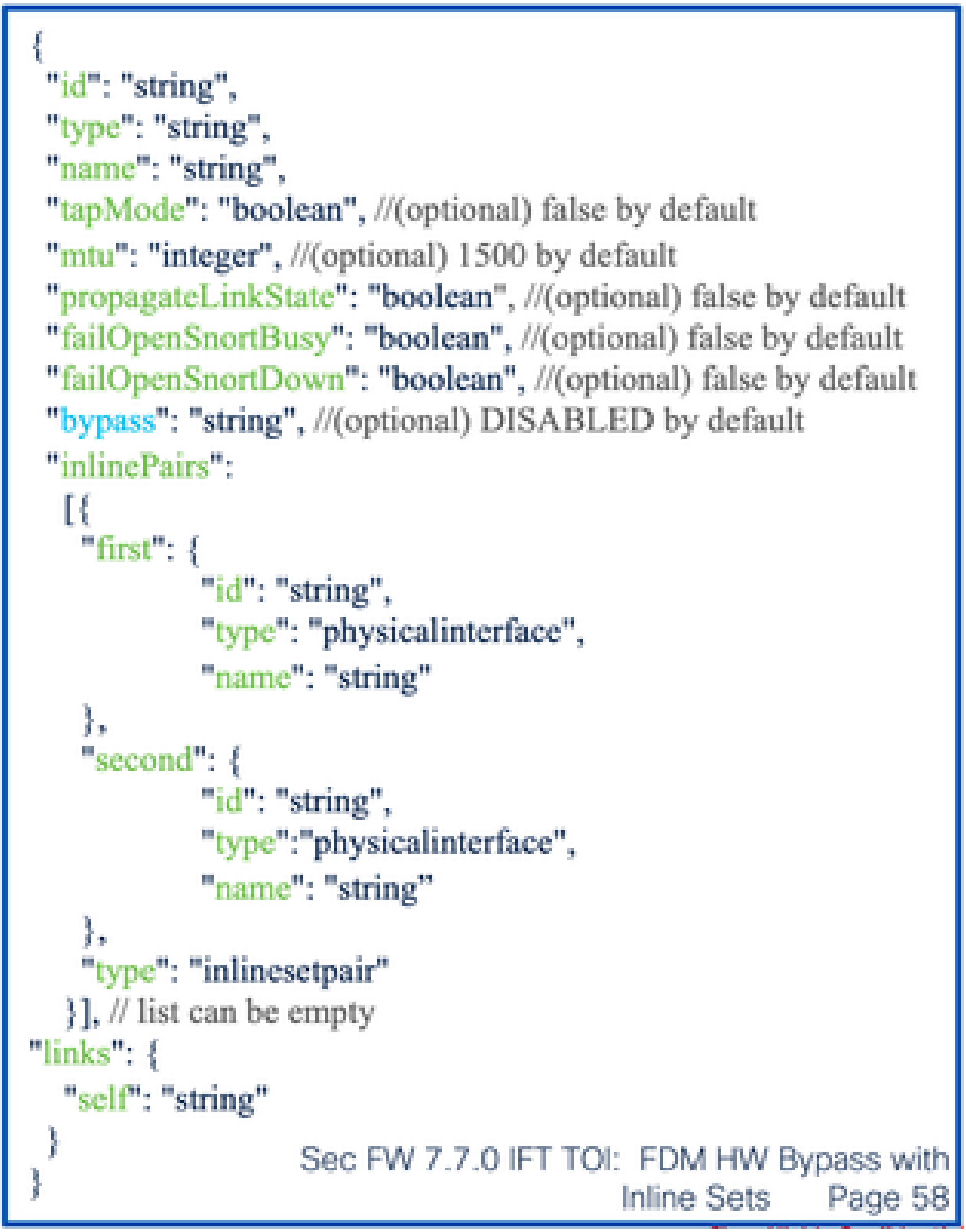
```
{ "interfaceInfoList": [
   {
      "interfaceId": "da9edc2d-58ba-11ef-b764-ffea0b8d9fa2",
      "hardwareName": "Ethernet1/4",
      "bypassInterfacePeerId": null,
      ...
   },
  {
      "interfaceId": "dbe9d2c1-58ba-11ef-b764-396644d1c752",
      "hardwareName": "Ethernet2/1",
      "bypassInterfacePeerId": "dc74fbc3-58ba-11ef-b764-11d423dbcbd7",
      ...
   },
   {
      "interfaceId": "dc74fbc3-58ba-11ef-b764-11d423dbcbd7",
      "hardwareName": "Ethernet2/2",
      "bypassInterfacePeerId": "dbe9d2c1-58ba-11ef-b764-396644d1c752",
      ...
   }],
  "id": "default",
  "type": "interfaceinfo",
  "links": { "self": "https://u90c04p02-
vrouter.cisco.com:25455/api/fdm/v6/operational/interfaceinfo/1/default"
   }
}
```

Interface Info REST API Example

Note: This is a snippet from the full call, due to size.

Inline Set REST APIs Model

- The Inline Set model consists of:
  - Type
  - Name
  - Tap Mode
  - MTU
  - Propagate Link State
  - Fail Open Snort Busy
  - Bypass values: DISABLED, STANDBY, BYPASS_FORCE



```
{
  "id": "string",
  "type": "string",
  "name": "string",
  "tapMode": "boolean", //(optional) false by default
  "mtu": "integer", //(optional) 1500 by default
  "propagateLinkState": "boolean", //(optional) false by default
  "failOpenSnortBusy": "boolean", //(optional) false by default
  "failOpenSnortDown": "boolean", //(optional) false by default
  "bypass": "string", //(optional) DISABLED by default
  "inlinePairs":
   [{
    "first": {
                 "id": "string",
                 "type": "physicalinterface",
                 "name": "string"
    },
    "second": {
                 "id": "string",
                 "type":"physicalinterface",
                 "name": "string"
    },
    "type": "inlinesetpair"
   }], // list can be empty
"links": {
   "self": "string"
  }
}
```

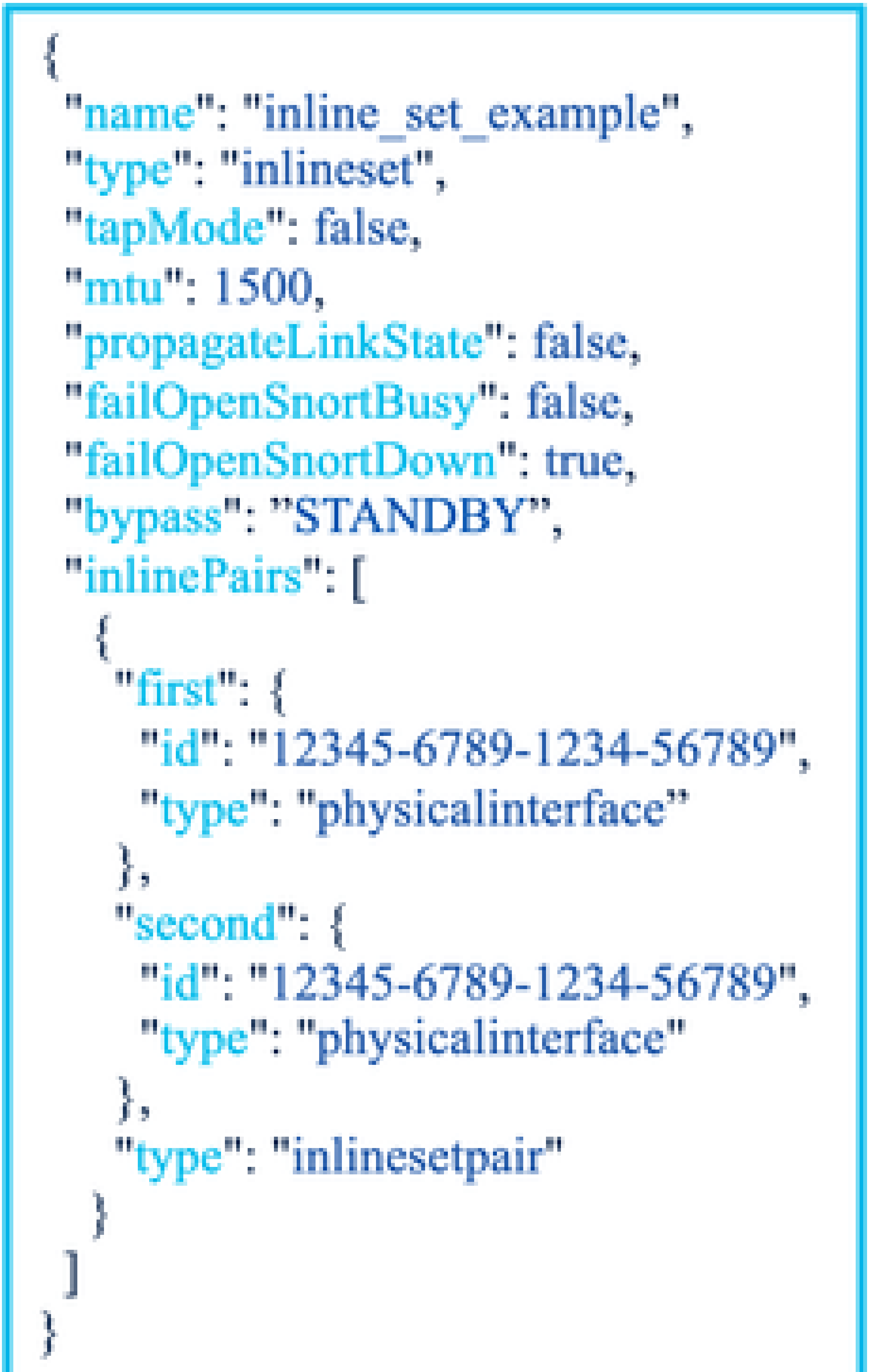
Sec FW 7.7.0 IFT TOI: FDM HW Bypass with Inline Sets Page 58

Inline Set REST API

Inline Set REST API Example

- Basic Inline Set examples with:
  - One Inline Pair

- Bypass Standby



```
{
  "name": "inline_set_example",
  "type": "inlineset",
  "tapMode": false,
  "mtu": 1500,
  "propagateLinkState": false,
  "failOpenSnortBusy": false,
  "failOpenSnortDown": true,
  "bypass": "STANDBY",
  "inlinePairs": [
    {
      "first": {
        "id": "12345-6789-1234-56789",
        "type": "physicalinterface"
      },
      "second": {
        "id": "12345-6789-1234-56789",
        "type": "physicalinterface"
      },
      "type": "inlinesetpair"
    }
  ]
}
```

2.Create Inline Set (see API Explorer for payload examples).

POST/devices/default/inlinesets
3.Create Security Zone (see API Explorer for payload examples) (optional).
POST/object/securityzones
4.Deploy to device (see API Explorer for payload examples).
POST/operational/deploy

Configure and Deploy an Inline Set with Hardware Bypass

1.Get interface IDs and information about Hardware Bypass interface pairs (see API Explorer for payload examples).
GET/operational/interfaceinfo/{objId}
2.Create Inline Set (see API Explorer for payload examples).
POST/devices/default/inlinesets
3.Create Security Zone (see API Explorer for payload examples) (optional).
POST/object/securityzones
4.Deploy to device (see API Explorer for payload examples).
POST/operational/deploy

Edit an Inline Set

1. Get interface IDs (see API Explorer for payload examples).
GET/devices/default/interfaces
2. Get Inline Sets.
GET/devices/default/inlinesets
3. Edit the Inline Set (see API Explorer for payload examples).
PUT/devices/default/inlinesets/{objId}
4. Deploy to device (see API Explorer for payload examples).
POST/operational/deploy

Verify

<#root>

```
> show running-config inline-set



inline-set test_inline_0
    interface-pair test2 test1
inline-set test_inline_1


hardware-bypass standby


    interface-pair test27 test28
inline-set test_inline_2
    hardware-bypass bypass
    interface-pair test26 test25


> show inline-set
```

```
 Inline-set test_inline_0
   Mtuis 1600 bytes
   Fail-open for snort down is off
   Fail-open for snort busy is off
   Tap mode is off
   Propagate-link-state option is off

hardware-bypass mode is disabled

   Interface-Pair[1]:
     Interface: Ethernet1/3 "test1"
       Current-Status: DOWN
     Interface: Ethernet1/4 "test2"
       Current-Status: DOWN
     Bridge Group ID: 519

> show inline-set

Inline-set test_inline_1
Mtuis 1500 bytes
Fail-open for snort down is off
Fail-open for snort busy is off
Tap mode is off
Propagate-link-state option is off
hardware-bypass mode is standby
Interface-Pair[1]:
Interface: Ethernet2/7 "test27"
Current-Status: DOWN
Interface: Ethernet2/8 "test28"
Current-Status: DOWN
Bridge Group ID: 618

> show inline-set

Inline-set test_inline_1
Mtuis 1500 bytes
Fail-open for snort down is off
Fail-open for snort busy is off
Tap mode is off
Propagate-link-state option is off

hardware-bypass mode is bypass

Interface-Pair[1]:
Interface: Ethernet2/6 "test26"
Current-Status: DOWN
Interface: Ethernet2/5 "test25"
Current-Status: DOWN
Bridge Group ID: 610

> show interface


...
```

```
Interface Ethernet1/7 "", is admin down, line protocol is down
Hardware is EtherSVI, BW 1000 Mbps, DLY 10 usec
Available but not configured via nameif

...
Interface Ethernet2/7 "", is admin down, line protocol is down
Hardware is EtherSVI, BW 1000 Mbps, DLY 10 usec

Hardware bypass is supported with interface Ethernet2/8


Available but not configured via nameif

...
Interface Ethernet2/8 "", is admin down, line protocol is down
Hardware is EtherSVI, BW 1000 Mbps, DLY 10 usec

Hardware bypass is supported with interface Ethernet2/7


Available but not configured via nameif
```

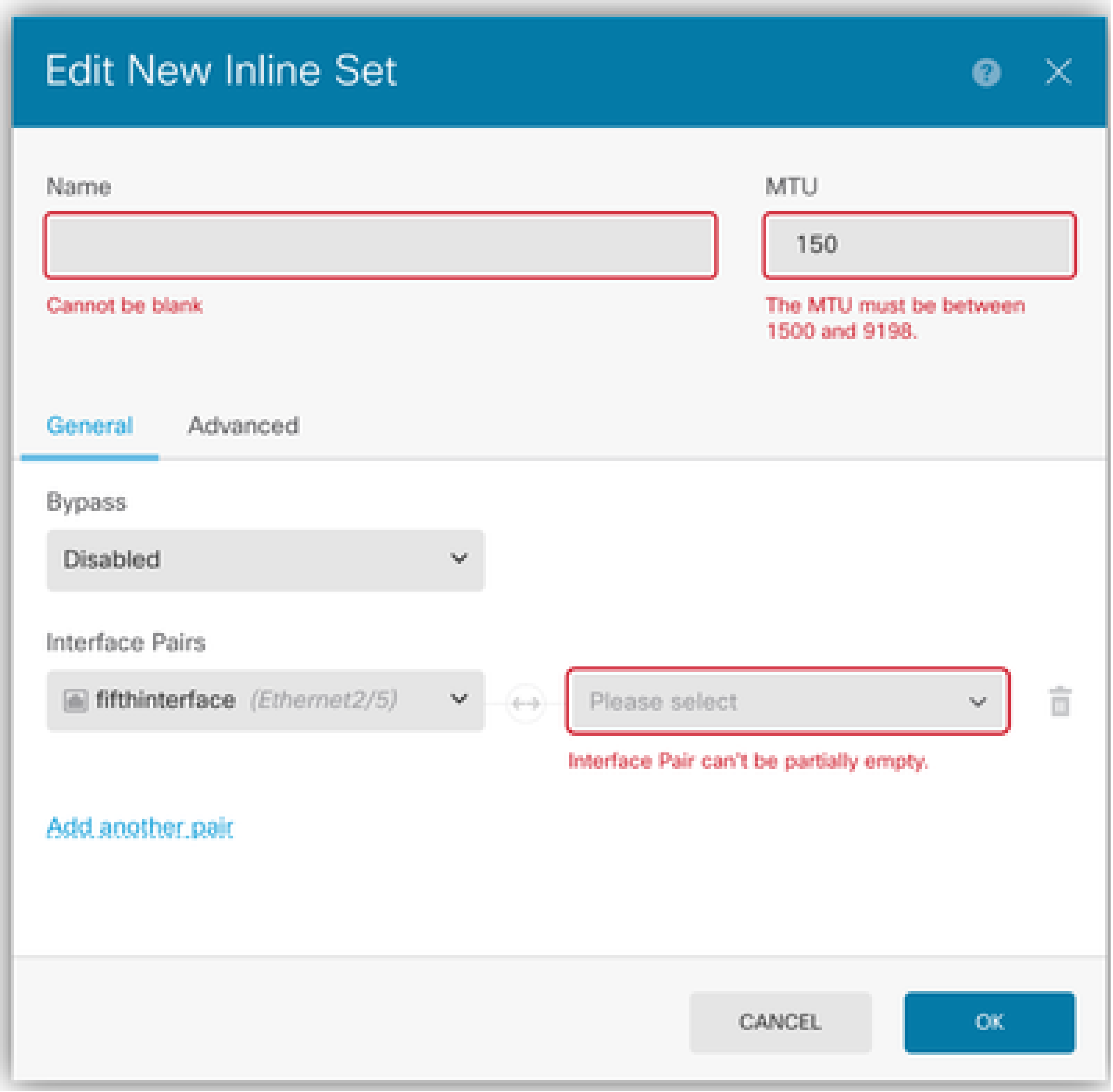
Troubleshoot

Commands

- **show running-config inline-set**
- **show inline-set**
- **show interface**
- **system support trace**

Inline Set - Validations When Creating

- Errors are presented on the GUI for each of the fields.
  - **Name** must be filled in.
  - The **MTU** size must be at least **1500.**
  - Both **interfaces** in a pair must be picked.



Edit New Inline Set

Name

Cannot be blank

MTU

150

The MTU must be between 1500 and 9198.

Bypass

Disabled

Interface Pairs

fifthinterface (Ethernet2/5)

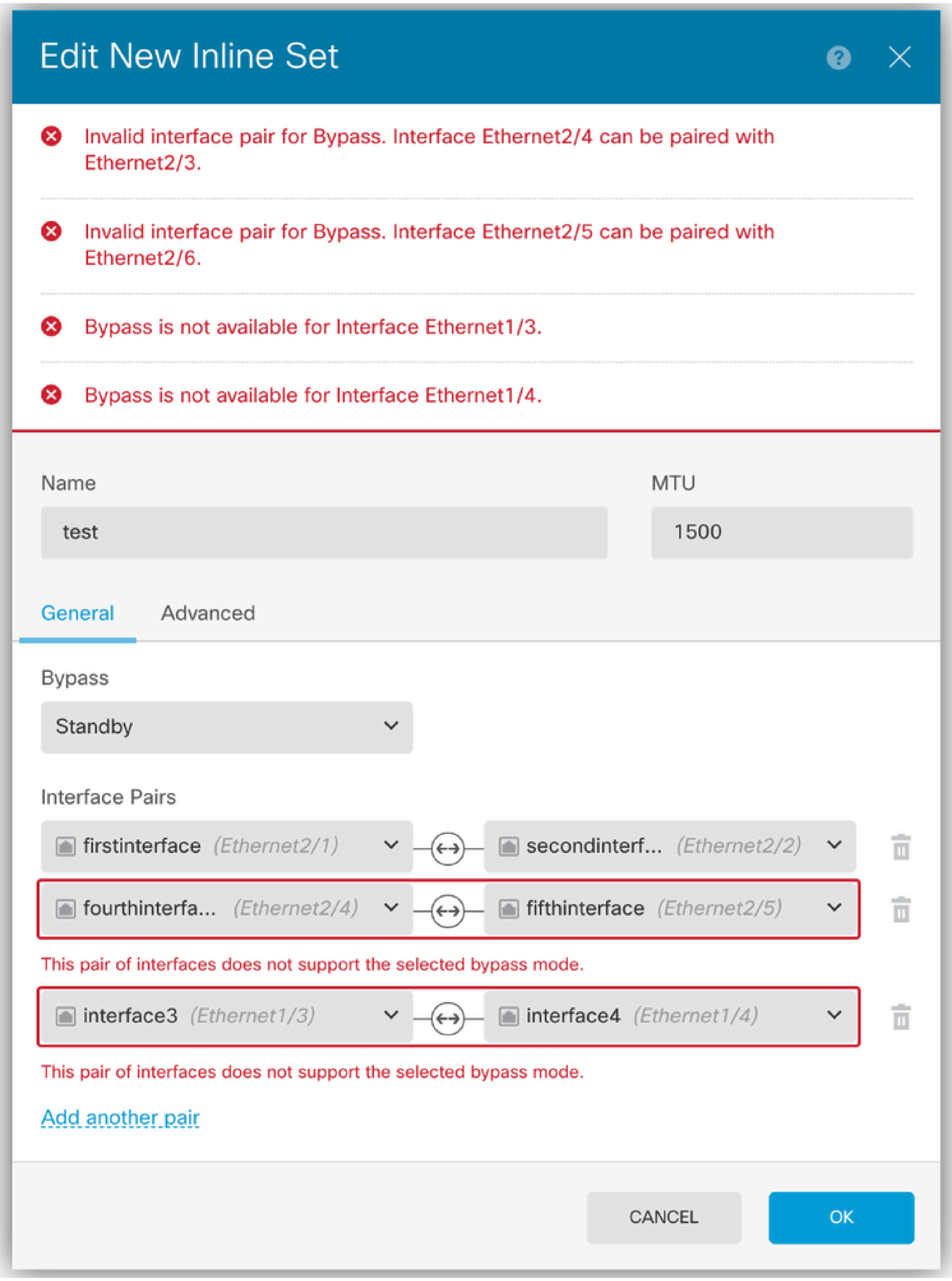
Please select

Interface Pair can't be partially empty.

MTU Size

Hardware Bypass - Validation When Creating

- New errors are presented on the GUI for each of the fields, when bypass is enabled:
  - All interfaces must support Bypass.
    - Error shows unsupported interfaces.
  - All pairs must use the predetermined interface pair.
    - Error message mentions available bypass interface pairs.



Edit New Inline Set

Invalid interface pair for Bypass. Interface Ethernet2/4 can be paired with Ethernet2/3.

Invalid interface pair for Bypass. Interface Ethernet2/5 can be paired with Ethernet2/6.

Bypass is not available for Interface Ethernet1/3.

Bypass is not available for Interface Ethernet1/4.

Name: test

MTU: 1500

Bypass: Standby

Interface Pairs

firstinterface (Ethernet2/1) ↔ secondinterf... (Ethernet2/2)

fourthinterfa... (Ethernet2/4) ↔ fifthinterface (Ethernet2/5)

This pair of interfaces does not support the selected bypass mode.

interface3 (Ethernet1/3) ↔ interface4 (Ethernet1/4)

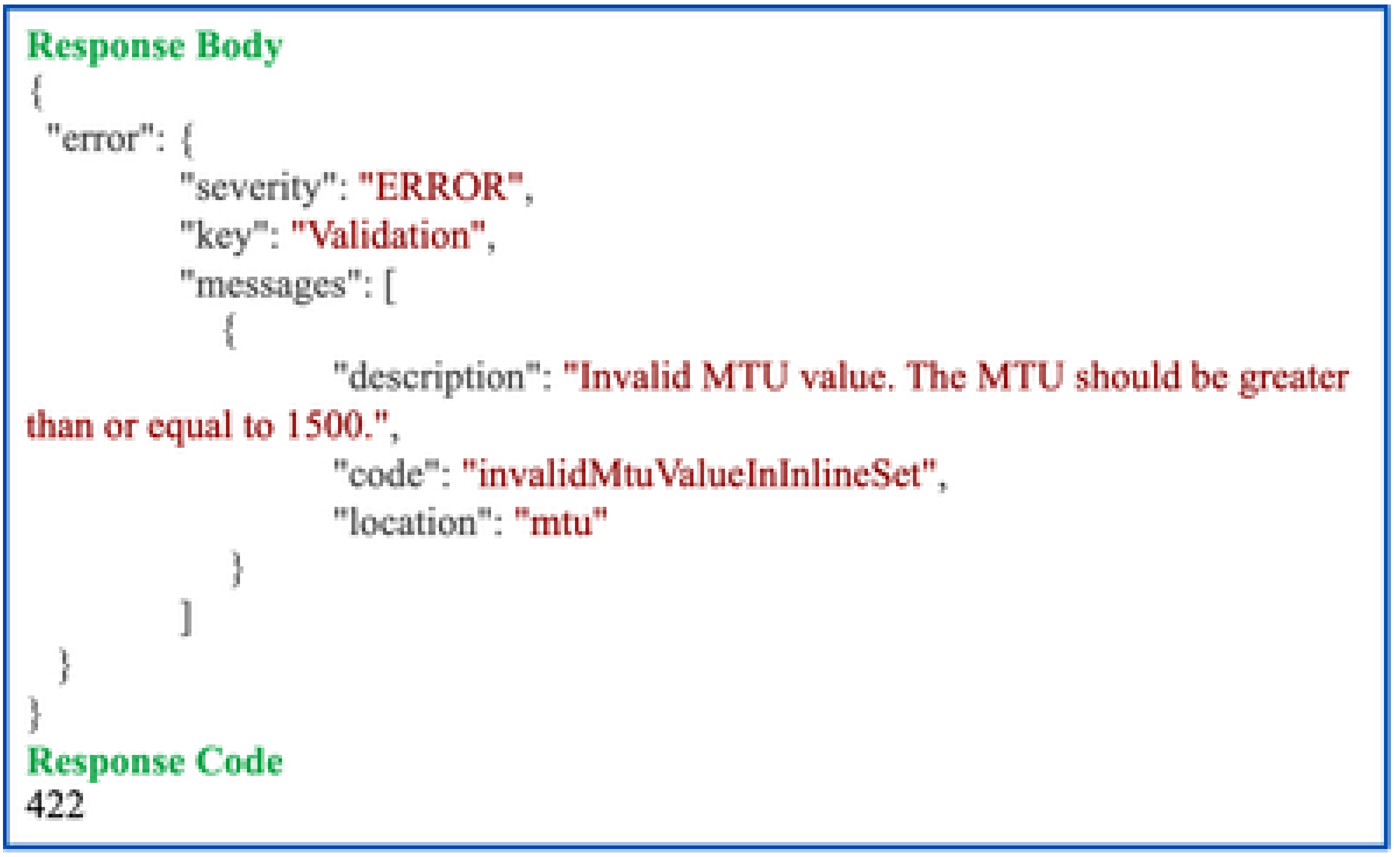
This pair of interfaces does not support the selected bypass mode.

GUI Validation

Note: The first pair (Ethernet2/1-Ethernet2/2) is valid.

REST API Response Shows Errors

- Errors are presented in the REST API response.
  - Here, the MTU value is invalid.



Response Body

```
{
 "error": {
            "severity": "ERROR",
            "key": "Validation",
            "messages": [
               {
                      "description": "Invalid MTU value. The MTU should be greater
than or equal to 1500.",
                      "code": "invalidMtuValueInInlineSet",
                      "location": "mtu"
               }
            ]
  }
}
```

Response Code

422

REST API Validation

Limitations of the Implementation for this Release

- Inline Sets: Works only with Physical Interfaces and EtherChannel.
- Inline Sets with Hardware Bypass: Works only with Physical Interfaces and requires a network module.

Unsupported Firewall Features on Inline Interfaces

- DHCP server
- DHCP relay
- DHCP client
- TCP Intercept
- Routing
- NAT
- VPN
- Application
- inspection
- QoS
- NetFlow

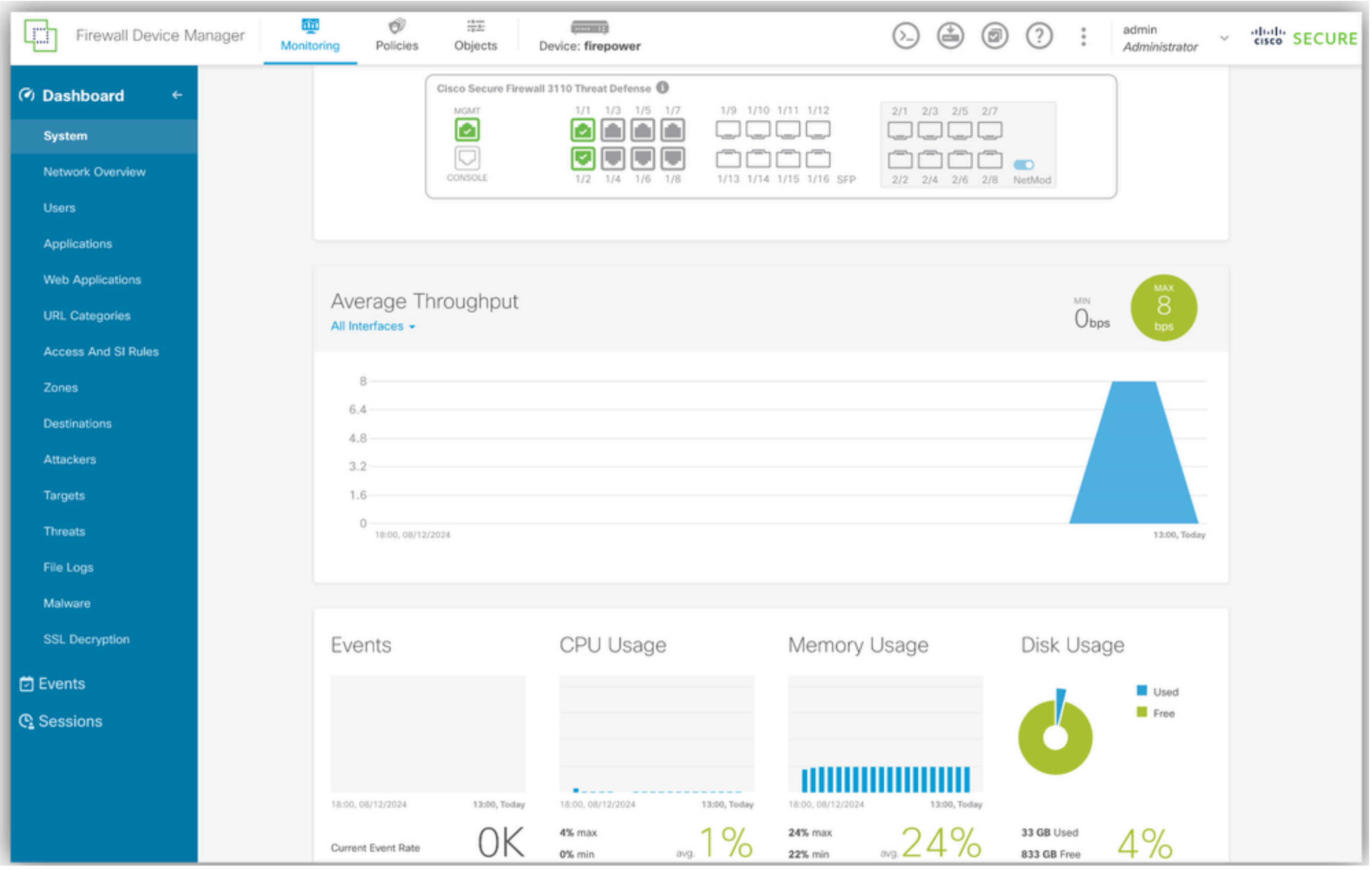
Verify Logs from CLI

- Logging.
  - Logs can be found in /ngfw/var/log/cisco/ngfw-onbox.log.

  - Search for Inline Set.
  - Example of possible errors found in logs:
    - Two interfaces do not support bypass.
    - Two interfaces are not a valid bypass pair.

```
root@FPR-3110-Pair:/home/admin# cd /ngfw/var/log/cisco/

root@FPR-3110-Pair:/ngfw/var/1og/cisco# cat ngfw-onbox.log | grep "InlineSet"

2024-08-28 12:35:00 ajp-nio-8009-exec-1: ERROR InlineSetValidator: 548 - Invalid

interface pair for Bypass. Interface Ethernet2/4 can be paired with Ethernet2/3.

2024-08-28 12:35:00 ajp-nio-8009-exec-1: ERROR InlineSetValidator:548 - Invalid

interface pair for Bypass. Interface Ethernet2/5 can be paired with Ethernet2/6.

2024-08-28 12:35:00 ajp-nio-8009-exec-1: ERROR InlineSetValidator:541 - Bypass

is not available for Interface Ethernet1/3.

2024-08-28 12:35:00 ajp-nio-8009-exec-1: ERROR InlineSetValidator:541 - Bypass

is not available for Interface
```

- Verify traffic from GUI.
  - Events are presented on the GUI.
  - Correctness of the traffic flow can be monitored here.
  - Navigate to **Monitoring > System.**



FDM Monitoring

- Verify traffic correctness from CLI.

<#root>

```
> system support trace
Enable firewall-engine-debug too? [n]:
Please specify an IP protocol: ICMP
Please specify a client IP address:
Please specify a server IP address:
Monitoring packet tracer debug messages
```

[packets show up here]

FAQs

Q: Is HA supported with inline-sets on FDM?
A: Inline Sets without Bypass are supported.
Inline Sets with Bypass are NOT supported.
Q: Are the spanning-tree BPDUs blocked on the inline-set pair?
A: No, they are not blocked.
Q: Are FTW cards supported in 3100?
A: Yes, FTW netmods have been supported since the 3100 Series was introduced with 7.1/9.17. Hardware Bypass is available starting 7.7.0.

Q: For 3100 FTW cards, is Bypass modes of Disabled, Standby, Bypass-Force like on FMC supported or not?
A: Hardware Bypass is available starting 7.7.0 on 3100 devices with FTW cards.
Q: Are Inline-Sets with port channels supported where the traffic is asymmetric across the port-channels as well?
A: No validation is performed on the PortChannel configured speed, so as long as the FTD supports it, it must be supported.
Q: In the event Snort fails for inspection, is failopen supported?
A: Please see the documentation on this setting on Firepower Management Center Configuration Guide.

Related Information

- Configure FTD Interfaces in Inline-Pair Mode
- Firepower Management Center Configuration Guide, Version 6.3
- Cisco Secure Firewall 3100 Series Hardware Installation Guide
- Cisco Secure Firewall 3100 Series Data Sheet