Configure FTD Interfaces in Inline-Pair Mode

Contents

Introduction

Prerequisites

- Requirements
- Components Used
- Related Products

Background Information

Configure Inline Pair Interface on FTD

- Network Diagram

Verify

Verify FTD Inline Pair Interface Operation

- Basic Theory
- Verification 1. With the Use of Packet-Tracer
- Verification 2. Send TCP SYN/ACK Packets Through Inline Pair
- Verification 3. Firewall Engine Debug For Allowed Traffic
- Verification 4. Verify Link-State Propagation
- Verification 5. Configure Static NAT
- Case Study - Asymmetric traffic through inline sets

Block Packet on Inline Pair Interface Mode

Configure Inline Pair Mode With Tap

Verify FTD Inline Pair With Tap Interface Operation

Inline Pair and Etherchannel

- Etherchannel Terminated on FTD
- Etherchannel through the FTD

Troubleshoot

Comparison: Inline Pair vs Inline Pair with Tap

Summary

Related Information

Introduction

This document describes the configuration, verification, and operation of an Inline Pair Interface on a Firepower Threat Defense (FTD) appliance.

Prerequisites

Requirements

There are no specific requirements for this document.

Components Used

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

- Firepower 4112 FTD (code 7.x)
- Firepower Management Center (FMC) (code 7.x)

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.

Related Products

This document can also be used with these hardware and software versions:

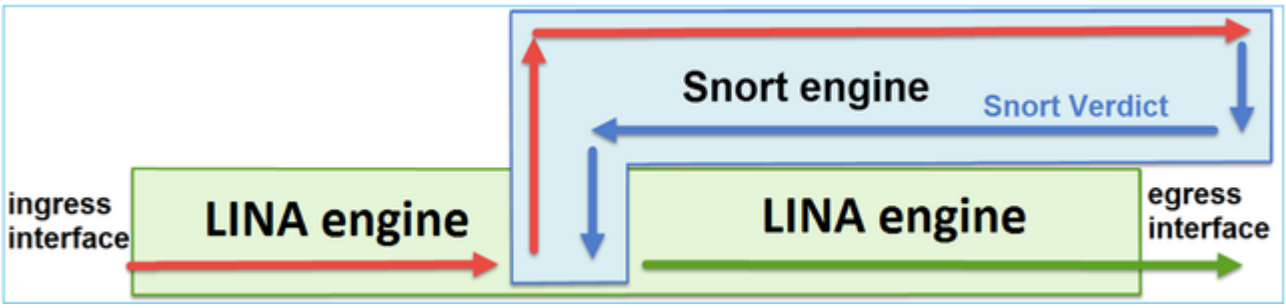
- FPR1000, FPR2100, FPR4100, FPR9300
- Secure Firewall 3100 and 4200 series
- vFTD
- FTD software code 6.2.x and later

Background Information

FTD is a unified software image that consists of 2 main engines:

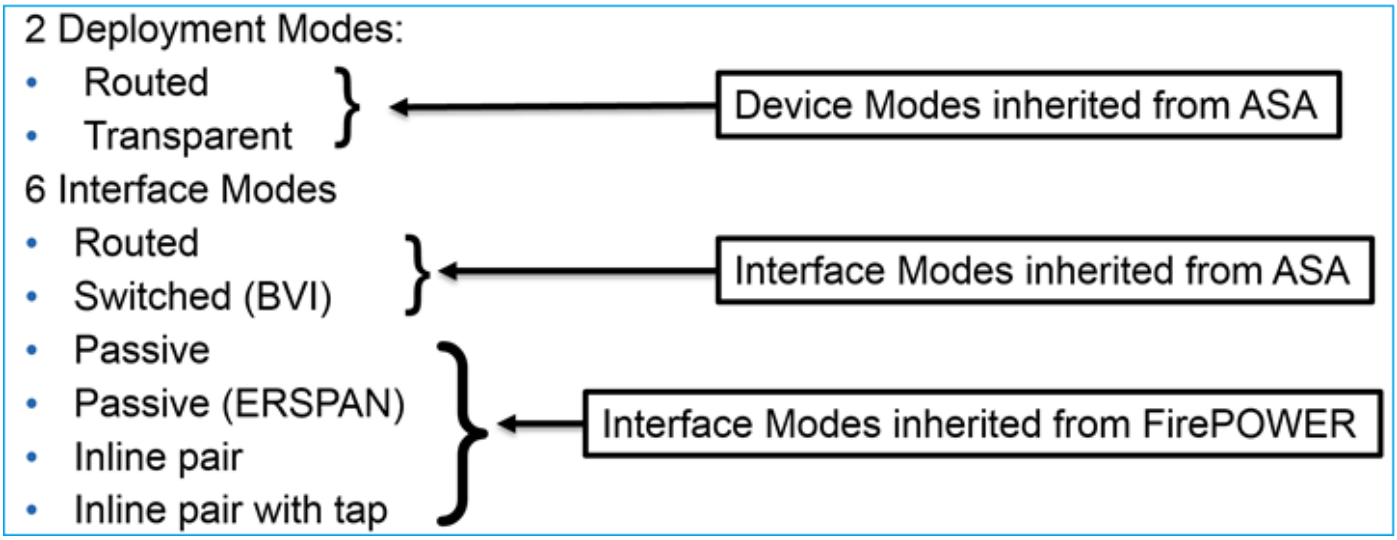
1. LINA engine
2. Snort engine

This figure shows how the 2 engines interact:



- A packet enters the ingress interface and it is handled by the LINA engine.
- If it is required by the FTD policy, the packet is inspected by the Snort engine.
- The Snort engine returns a verdict for the packet.
- The LINA engine drops or forwards the packet based on Snort's verdict.

FTD provides two Deployment modes and six Interface modes as shown in the image:



Note: You can mix interface modes on a single FTD appliance.

Here is a high level overview of the various FTD deployment and interface modes:

FTD Interface Mode	FTD Deployment Mode	Description	Traffic can be Dropped
Routed	Routed	Full LINA-engine and Snort-engine checks.	Yes
Switched	Transparent	Full LINA-engine and Snort-engine checks.	Yes
Inline Pair	Routed or	Partial LINA-engine and full Snort-	Yes

	Transparent	engine checks.	
Inline Pair with Tap	Routed or Transparent	Partial LINA-engine and full Snort-engine checks.	No
Passive	Routed or Transparent	Partial LINA-engine and full Snort-engine checks.	No
Passive (ERSPAN)	Routed	Partial LINA-engine and full Snort-engine checks.	No

Configure Inline Pair Interface on FTD

Network Diagram



Requirement

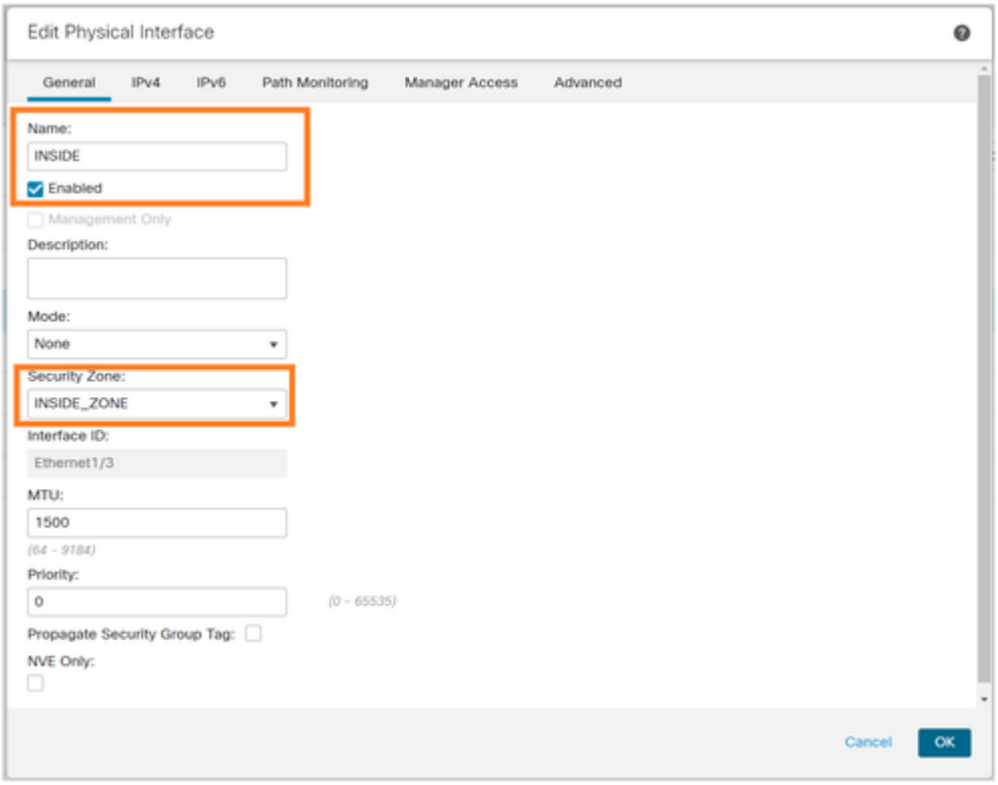
Configure physical interfaces e1/3 and e1/4 in Inline Pair Mode as per these requirements:

Interface	e1/3	e1/4
Name	INSIDE	OUTSIDE
Security Zone	INSIDE_ZONE	OUTSIDE_ZONE
Inline Set name	Inline-Pair-1	
Inline Set MTU	1500	
Propagate Link State	Enabled	

Solution

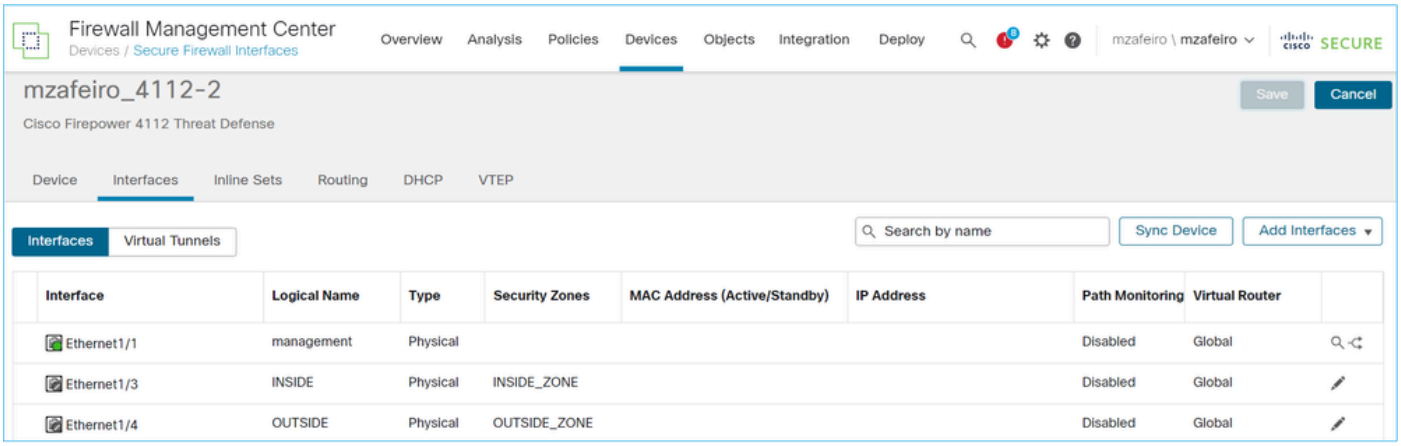
Step 1. In order to configure to the individual interfaces, navigate to **Devices > Device Management**, select the appropriate **device** and select **Edit**:

Next, specify **Name** and tick **Enabled** for the interface as shown in the image.



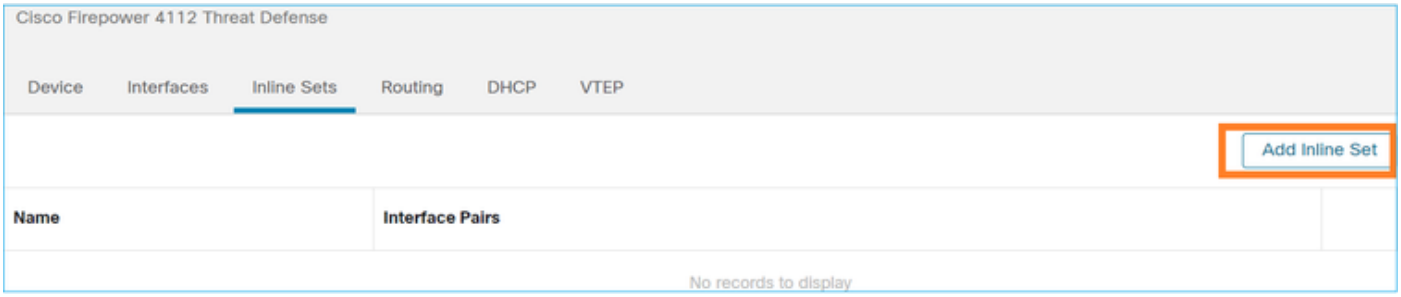
Note: The Name is the the name of the interface.

Similarly, for interface Ethernet1/4. The final result is shown:

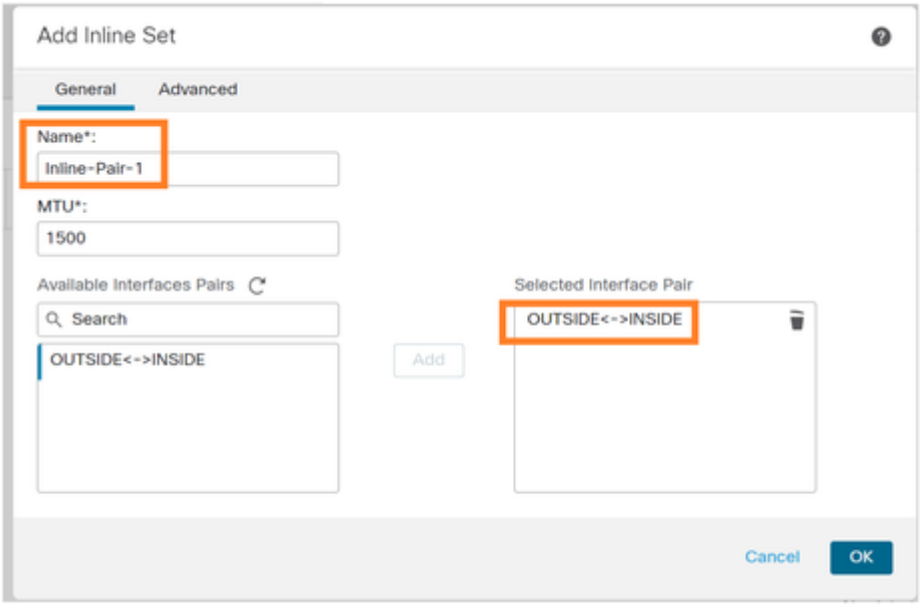


Step 2. Configure the Inline Pair.

Navigate to **Inline Sets > Add Inline Set** as shown in the image.



Step 3. Configure the **General settings** as per the requirements as shown in the image.



Step 4. Enable **Propagate Link State** option in the **Advanced Settings** as shown in the image.



Link state propagation automatically brings down the second interface in the inline interface pair when one of the interfaces in the inline set goes down.

Step 5. **Save** the changes and **Deploy**.

Verify

Use this section in order to confirm that your configuration works properly.

Verify the Inline Pair configuration from the FTD CLI.

Solution

Log in to FTD CLI and verify the Inline Pair configuration:

<#root>

```
firepower#
```

show inline-set

```
Inline-set Inline-Pair-1
  Mtu is 1500 bytes
  Fail-open for snort down is on
  Fail-open for snort busy is off
  Tap mode is off
  Propagate-link-state option is on
  hardware-bypass mode is disabled
  Interface-Pair[1]:
    Interface: Ethernet1/4 "OUTSIDE"
      Current-Status: UP
    Interface: Ethernet1/3 "INSIDE"
      Current-Status: UP
    Bridge Group ID: 507
```

Note: The Bridge Group ID is a value different than 0. If Tap Mode is on, then it is 0.

Interface and name information:

<#root>

```
firepower#
```

show nameif

```
Interface       Name            Security
```

```
Ethernet1/1     management        0
Ethernet1/3     INSIDE            0
Ethernet1/4     OUTSIDE           0
```

Verify the interface status:

<#root>

```
firepower#
```

show interface ip brief

```
Interface              IP-Address    OK? Method Status Protocol
Internal-Control0/0    unassigned    YES unset  up     up
Internal-Data0/0       unassigned    YES unset  up     up
Internal-Data0/1       unassigned    YES unset  up     up
Internal-Data0/2       169.254.1.1   YES unset  up     up
Internal-Data0/3       unassigned    YES unset  up     up
Internal-Data0/4       unassigned    YES unset  down   up
Ethernet1/1            203.0.113.130 YES unset  up     up
```

Ethernet1/3 unassigned YES unset up up

Ethernet1/4 unassigned YES unset up up

Verify physical interface information:

<#root>

```
firepower#
```

show interface e1/3

Interface Ethernet1/3 "INSIDE", is up, line protocol is up

```
  Hardware is EtherSVI, BW 1000 Mbps, DLY 10 usec
           MAC address ac4a.670e.641e, MTU 1500
```

IPS Interface-Mode: inline, Inline-Set: Inline-Pair-1

```
        IP address unassigned
Traffic Statistics for "INSIDE":
        170 packets input, 12241 bytes
         41 packets output, 7881 bytes
          9 packets dropped
    1 minute input rate 0 pkts/sec, 37 bytes/sec
    1 minute output rate 0 pkts/sec, 19 bytes/sec
    1 minute drop rate, 0 pkts/sec
    5 minute input rate 0 pkts/sec, 34 bytes/sec
    5 minute output rate 0 pkts/sec, 23 bytes/sec
    5 minute drop rate, 0 pkts/sec
```

Verify FTD Inline Pair Interface Operation

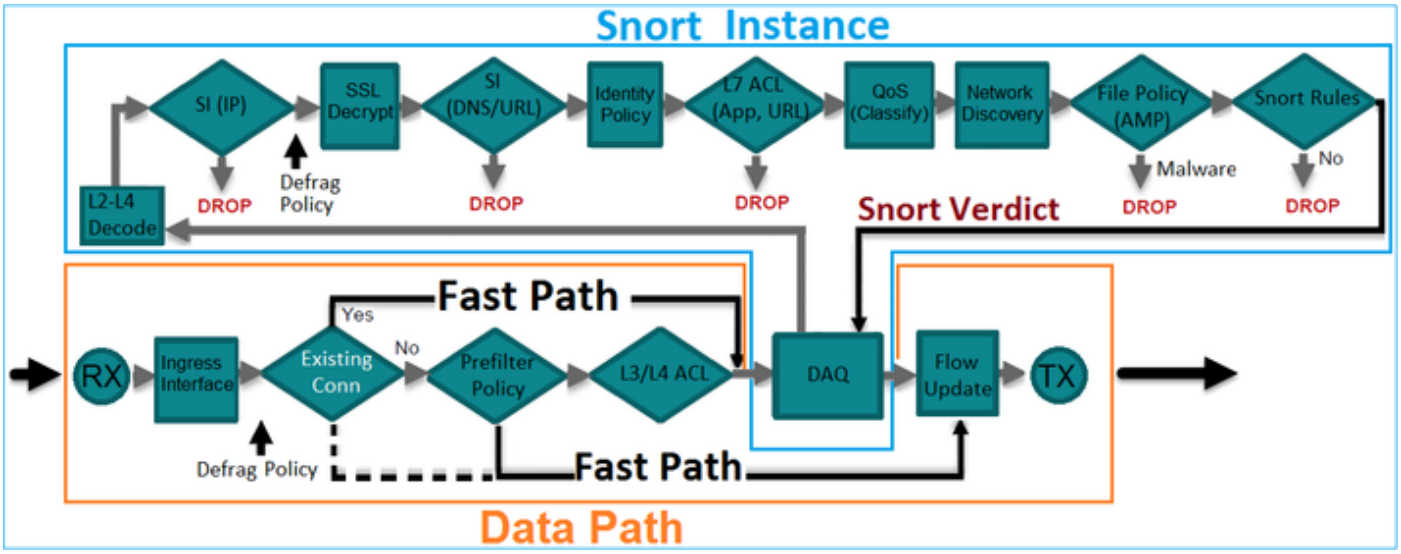
This section covers these verification checks in order to verify the Inline Pair operation:

- Verification 1. With the use of packet-tracer.
- Verification 2. Enable capture with trace and send a TCP synchronize/acknowledge (SYN/ACK) packet through the Inline Pair.
- Verification 3. Monitor FTD traffic with the use of firewall engine debug
- Verification 4. Verify the Link-State Propagation functionality.
- Verification 5. Configure Static Network Address Translation (NAT).

Solution

Architectural Overview

When 2 FTD interfaces operate in Inline-pair mode, a packet is handled, as shown in the image.

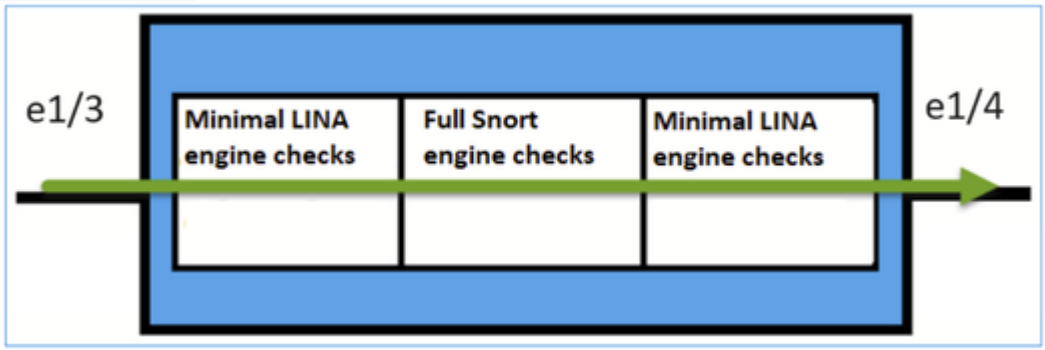


Note: Only physical interfaces can be members of an Inline pair set.

Basic Theory

- When you configure an Inline Pair 2 Physical, interfaces are internally bridged.
- Very similar to classic inline Intrusion Prevention System (IPS).
- Available in Routed or Transparent Deployment modes.
- Most of the LINA engine features (NAT, Routing, and so on) are not available for flows which go through an Inline Pair.
- Transit traffic can be dropped.
- A few LINA engine checks are applied along with full Snort engine checks.

The last point can be visualized as shown in the image:



Verification 1. With the Use of Packet-Tracer

The packet-tracer output which emulates a packet that traverses the inline pair with the important points highlighted:

<#root>

```
firepower#
```

packet-tracer input INSIDE tcp 192.168.201.50 1111 192.168.202.50 80

Phase: 1

Type: NGIPS-MODE

Subtype: ngips-mode

Result: ALLOW

Elapsed time: 11834 ns

Config:

Additional Information:

The flow ingressed an interface configured for NGIPS mode and NGIPS services will be applied

Phase: 2
Type: ACCESS-LIST
Subtype:
Result: ALLOW
Elapsed time: 11834 ns
Config:
access-group CSM_FW_ACL_ global
access-list CSM_FW_ACL_ advanced permit ip host 192.168.201.50 host 192.168.202.50 rule-id 268451044
access-list CSM_FW_ACL_ remark rule-id 268451044: ACCESS POLICY: mzafeiro_2m - Mandatory
access-list CSM_FW_ACL_ remark rule-id 268451044: L7 RULE: New-Rule-#1303-ALLOW
Additional Information:
This packet will be sent to snort for additional processing where a verdict will be reached

Phase: 3

Type: NGIPS-EGRESS-INTERFACE-LOOKUP

Subtype: Resolve Egress Interface

Result: ALLOW

Elapsed time: 2440 ns

Config:

Additional Information:

Ingress interface INSIDE is in NGIPS inline mode.

Egress interface OUTSIDE is determined by inline-set configuration

Phase: 4
Type: FLOW-CREATION
Subtype:
Result: ALLOW
Elapsed time: 68320 ns
Config:
Additional Information:
New flow created with id 1801, packet dispatched to next module

Phase: 5
Type: EXTERNAL-INSPECT
Subtype:
Result: ALLOW
Elapsed time: 18056 ns
Config:
Additional Information:
Application: 'SNORT Inspect'

Phase: 6
Type: SNORT
Subtype: identity
Result: ALLOW
Elapsed time: 13668 ns
Config:
Additional Information:
user id: no auth, realm id: 0, device type: 0, auth type: invalid, auth proto: basic, username: none, A
src sgt: 0, src sgt type: unknown, dst sgt: 0, dst sgt type: unknown, abp src: none, abp dst: none, loc

Phase: 7
Type: SNORT
Subtype: firewall
Result: ALLOW
Elapsed time: 67770 ns
Config:
Network 0, Inspection 0, Detection 0, Rule ID 268451044
Additional Information:
Starting rule matching, zone -1 -> -1, geo 0 -> 0, vlan 0, src sgt: 0, src sgt type: unknown, dst sgt:
Matched rule ids 268451044 - Allow

```
Phase: 8
Type: SNORT
Subtype: appid
Result: ALLOW
Elapsed time: 11002 ns
Config:
Additional Information:
service: (0), client: (0), payload: (0), misc: (0)

Result:

input-interface: INSIDE(vrfid:0)



input-status: up



input-line-status: up



output-interface: OUTSIDE(vrfid:0)



output-status: up



output-line-status: up



Action: allow


Time Taken: 204924 ns
```

Verification 2. Send TCP SYN/ACK Packets Through Inline Pair

You can generate TCP SYN/ACK packets with the use of a packet that crafts utility like Scapy. This syntax generates 3 packets with SYN/ACK flags enabled:

<#root>

```
root@KALI:~#

scapy

INFO: Can't import python gnuplot wrapper . Won't be able to plot.
WARNING: No route found for IPv6 destination :: (no default route?)
Welcome to Scapy (2.2.0)
>>>

conf.iface='eth0'

>>>

packet = IP(dst="192.168.201.60")/TCP(flags="SA",dport=80)

>>>

syn_ack=[]

>>>

for i in range(0,3): # Send 3 packets

...

syn_ack.extend(packet)

...
>>>

send(syn_ack)
```

Enable this capture on FTD CLI and send a few TCP SYN/ACK packets:

<#root>

```
firepower#

capture CAPI interface INSIDE trace match ip host 192.168.201.60 any


firepower#

capture CAPO interface OUTSIDE match ip host 192.168.201.60 any
```

The captures shows that the 3 SYN/ACK packets traverse the FTD:

<#root>

firepower#

show capture CAPI

3 packets captured

1: 09:20:18.206440 192.168.201.50.20 > 192.168.201.60.80: S 0:0(0) ack 0 win 8192
2: 09:20:18.208180 192.168.201.50.20 > 192.168.201.60.80: S 0:0(0) ack 0 win 8192
3: 09:20:18.210026 192.168.201.50.20 > 192.168.201.60.80: S 0:0(0) ack 0 win 8192
3 packets shown
firepower#

show capture CAPO

3 packets captured

1: 09:20:18.206684 192.168.201.50.20 > 192.168.201.60.80: S 0:0(0) ack 0 win 8192
2: 09:20:18.208210 192.168.201.50.20 > 192.168.201.60.80: S 0:0(0) ack 0 win 8192
3: 09:20:18.210056 192.168.201.50.20 > 192.168.201.60.80: S 0:0(0) ack 0 win 8192
3 packets shown

The Trace of the first capture packet reveals some additional information like the Snort engine verdict:

<#root>

firepower#

show capture CAPI packet-number 1 trace

3 packets captured

1: 09:20:18.206440 192.168.201.50.20 > 192.168.201.60.80: S 0:0(0) ack 0 win 8192
Phase: 1
Type: NGIPS-MODE
Subtype: ngips-mode
Result: ALLOW
Elapsed time: 5978 ns
Config:
Additional Information:
The flow ingressed an interface configured for NGIPS mode and NGIPS services will be applied

Phase: 2
Type: ACCESS-LIST
Subtype:
Result: ALLOW
Elapsed time: 5978 ns
Config:
access-group CSM_FW_ACL_ global
access-list CSM_FW_ACL_ advanced permit ip host 192.168.201.50 object-group FMC_INLINE_dst_rule_2684510

access-list CSM_FW_ACL_ remark rule-id 268451044: ACCESS POLICY: mzafeiro_2m - Mandatory
access-list CSM_FW_ACL_ remark rule-id 268451044: L7 RULE: New-Rule-#1303-ALLOW
object-group network FMC_INLINE_dst_rule_268451044
network-object 192.168.202.50 255.255.255.255
network-object 192.168.201.60 255.255.255.255
Additional Information:
This packet will be sent to snort for additional processing where a verdict will be reached

Phase: 3
Type: NGIPS-EGRESS-INTERFACE-LOOKUP
Subtype: Resolve Egress Interface
Result: ALLOW
Elapsed time: 1952 ns
Config:
Additional Information:
Ingress interface INSIDE is in NGIPS inline mode.
Egress interface OUTSIDE is determined by inline-set configuration

Phase: 4
Type: FLOW-CREATION
Subtype:
Result: ALLOW
Elapsed time: 45872 ns
Config:
Additional Information:
New flow created with id 1953, packet dispatched to next module

Phase: 5
Type: EXTERNAL-INSPECT
Subtype:
Result: ALLOW
Elapsed time: 18544 ns
Config:
Additional Information:
Application: 'SNORT Inspect'

Phase: 6
Type: SNORT
Subtype: identity
Result: ALLOW
Elapsed time: 25182 ns
Config:
Additional Information:
user id: no auth, realm id: 0, device type: 0, auth type: invalid, auth proto: basic, username: none, A
src sgt: 0, src sgt type: unknown, dst sgt: 0, dst sgt type: unknown, abp src: none, abp dst: none, loc

Phase: 7

Type: SNORT

Subtype: firewall

Result: ALLOW

Elapsed time: 50924 ns

Config:

Network 0, Inspection 0, Detection 0, Rule ID 268451044

Additional Information:

Starting rule matching, zone -1 -> -1, geo 0 -> 0, vlan 0, src sgt: 0, src sgt type: unknown, dst sgt: 0

Matched rule ids 268451044 - Allow

Phase: 8
Type: SNORT
Subtype: appid
Result: ALLOW
Elapsed time: 17722 ns
Config:
Additional Information:
service: (0), client: (0), payload: (0), misc: (0)

Result:
input-interface: INSIDE(vrfid:0)
input-status: up
input-line-status: up
output-interface: OUTSIDE(vrfid:0)
output-status: up
output-line-status: up
Action: allow
Time Taken: 172152 ns

1 packet shown

The trace of the second captured packet shows that the packet matches a current connection so it bypasses the ACL check, but still is inspected by the Snort engine:

<#root>

```
firepower#

show capture CAPI packet-number 2 trace



3 packets captured

2: 09:20:18.208180 192.168.201.50.20 > 192.168.201.60.80: S 0:0(0) ack 0 win 8192
Phase: 1
Type: FLOW-LOOKUP
Subtype:
Result: ALLOW
Elapsed time: 1952 ns
Config:
Additional Information:

Found flow with id 1953, using existing flow



Phase: 2
Type: EXTERNAL-INSPECT
Subtype:
Result: ALLOW
Elapsed time: 7320 ns
Config:
Additional Information:
Application: 'SNORT Inspect'

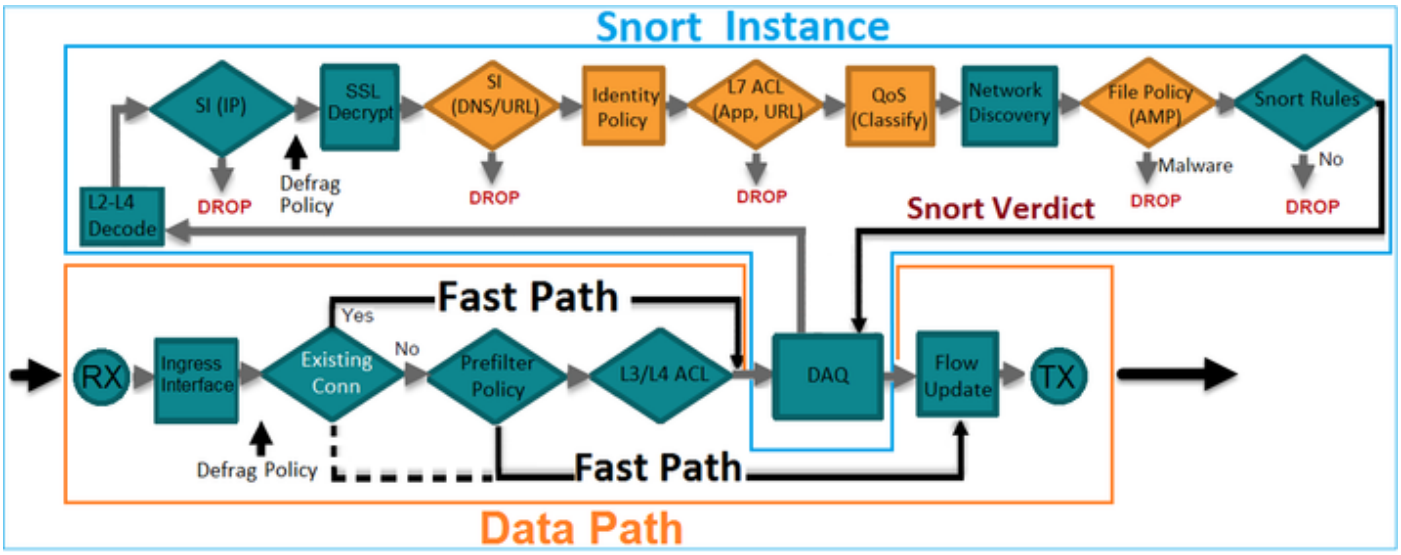
Phase: 3
Type: SNORT
Subtype: appid
Result: ALLOW
Elapsed time: 1860 ns
Config:
Additional Information:
service: (0), client: (0), payload: (0), misc: (0)

Result:
input-interface: INSIDE(vrfid:0)
input-status: up
input-line-status: up
Action: allow
Time Taken: 11132 ns


1 packet shown
```

Verification 3. Firewall Engine Debug For Allowed Traffic

Firewall engine debug runs against specific components of the FTD Snort Engine like the Access Control Policy, as shown in the image:



When you send the TCP SYN/ACK packets through Inline Pair, you can see in the debug output:

<#root>

>

system support firewall-engine-debug

Please specify an IP protocol:

tcp

Please specify a client IP address:
Please specify a client port:
Please specify a server IP address:

192.168.201.60

Please specify a server port:

80

Monitoring firewall engine debug messages

192.168.201.60-80 > 192.168.201.50-20 6 AS 4 I 12 New session

192.168.201.60-80 > 192.168.201.50-20 6 AS 4 I 12 using HW or preset rule order 3, id 268438528 action A

```
192.168.201.60-80 > 192.168.201.50-20 6 AS 4 I 12 allow action
```

```
192.168.201.60-80 > 192.168.201.50-20 6 AS 4 I 12 Deleting session
```

Verification 4. Verify Link-State Propagation

Enable buffer log on FTD and shutdown the switchport connected to e1/4 interface. On FTD CLI, you must see that both interfaces went down:

<#root>

```
firepower#
```

```
show interface ip brief
```

```
Interface              IP-Address    OK? Method Status     Protocol
Internal-Control0/0    unassigned    YES unset  up         up
Internal-Data0/0       unassigned    YES unset  up         up
Internal-Data0/1       unassigned    YES unset  up         up
Internal-Data0/2       169.254.1.1   YES unset  up         up
Internal-Data0/3       unassigned    YES unset  up         up
Internal-Data0/4       unassigned    YES unset  down       up
Ethernet1/1            203.0.113.130 YES unset  up         up
```

```
Ethernet1/3            unassigned    YES unset  admin down down
```

```
Ethernet1/4            unassigned    YES unset  down       down
```

The FTD logs show:

<#root>

```
firepower#
```

```
show log
```

```
...

May 28 2024 07:35:10: %FTD-4-411002: Line protocol on Interface Ethernet1/4, changed state to down



May 28 2024 07:35:10: %FTD-4-411004: Interface INSIDE, changed state to administratively down



May 28 2024 07:35:10: %FTD-4-411004: Interface Ethernet1/3, changed state to administratively down



May 28 2024 07:35:10: %FTD-4-812005: Link-State-Propagation activated on inline-pair due to failure of i



May 28 2024 07:35:10: %FTD-4-411002: Line protocol on Interface Ethernet1/3, changed state to down
```

The inline-set status shows the state of the 2 interface members:

```
<#root>

firepower#

show inline-set



Inline-set Inline-Pair-1
  Mtu is 1500 bytes
  Fail-open for snort down is on
  Fail-open for snort busy is off
  Tap mode is off


Propagate-link-state option is on


  hardware-bypass mode is disabled
    Interface-Pair[1]:
    Interface: Ethernet1/4 "OUTSIDE"


    Current-Status: Down(Propagate-Link-State-Activated)


    Interface: Ethernet1/3 "INSIDE"
```

Current-Status: Down(Administrative-Down-By-Propagate-Link-State)

Bridge Group ID: 507

Notice the difference in the status of the 2 interfaces:

<#root>

firepower#

show interface e1/3

Interface Ethernet1/3 "INSIDE", is admin down, line protocol is down

  Hardware is EtherSVI, BW 1000 Mbps, DLY 10 usec
        MAC address ac4a.670e.641e, MTU 1500
        IPS Interface-Mode: inline, Inline-Set: Inline-Pair-1

Administrative-Down-By-Propagate-Link-State

        IP address unassigned
  Traffic Statistics for "INSIDE":
        2400 packets input, 165873 bytes
        1822 packets output, 178850 bytes
        17 packets dropped
      1 minute input rate 0 pkts/sec, 0 bytes/sec
      1 minute output rate 0 pkts/sec, 0 bytes/sec
      1 minute drop rate, 0 pkts/sec
      5 minute input rate 0 pkts/sec, 32 bytes/sec
      5 minute output rate 0 pkts/sec, 57 bytes/sec
      5 minute drop rate, 0 pkts/sec
firepower#

show interface e1/4

Interface Ethernet1/4 "OUTSIDE", is down, line protocol is down

  Hardware is EtherSVI, BW 1000 Mbps, DLY 10 usec
        MAC address ac4a.670e.640e, MTU 1500
        IPS Interface-Mode: inline, Inline-Set: Inline-Pair-1

        Propagate-Link-State-Activated

```
       IP address unassigned
  Traffic Statistics for "OUTSIDE":
       1893 packets input, 158046 bytes
       2386 packets output, 213997 bytes
       67 packets dropped
     1 minute input rate 0 pkts/sec,  0 bytes/sec
     1 minute output rate 0 pkts/sec,  0 bytes/sec
     1 minute drop rate, 0 pkts/sec
     5 minute input rate 0 pkts/sec,  51 bytes/sec
     5 minute output rate 0 pkts/sec,  39 bytes/sec
     5 minute drop rate, 0 pkts/sec
```

After you re-enable the switchport, the FTD logs show:

<#root>

May 28 2024 07:38:04: %FTD-4-411001: Line protocol on Interface Ethernet1/4, changed state to up

May 28 2024 07:38:04: %FTD-4-411003: Interface Ethernet1/3, changed state to administratively up

May 28 2024 07:38:04: %FTD-4-411003: Interface INSIDE, changed state to administratively up

May 28 2024 07:38:04: %FTD-4-812006: Link-State-Propagation de-activated on inline-pair due to recovery

May 28 2024 07:38:05: %FTD-4-411002: Line protocol on Interface Ethernet1/4, changed state to down

Verification 5. Configure Static NAT

Solution

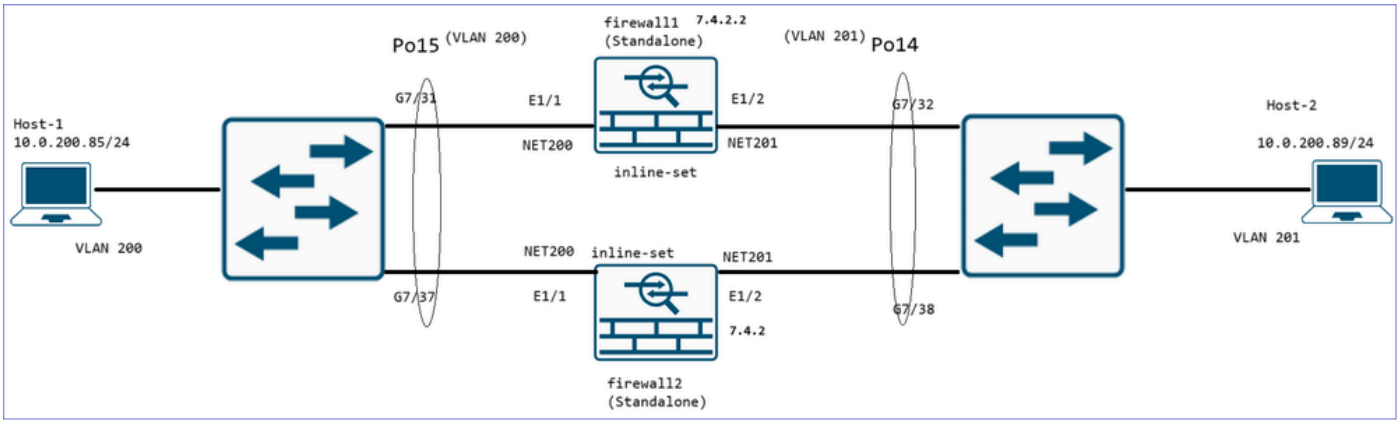
NAT is not supported for interfaces that operate in inline, inline tap, or passive modes:

Firepower Management Center Configuration Guide, Version 6.0.1

Case Study - Asymmetric traffic through inline sets

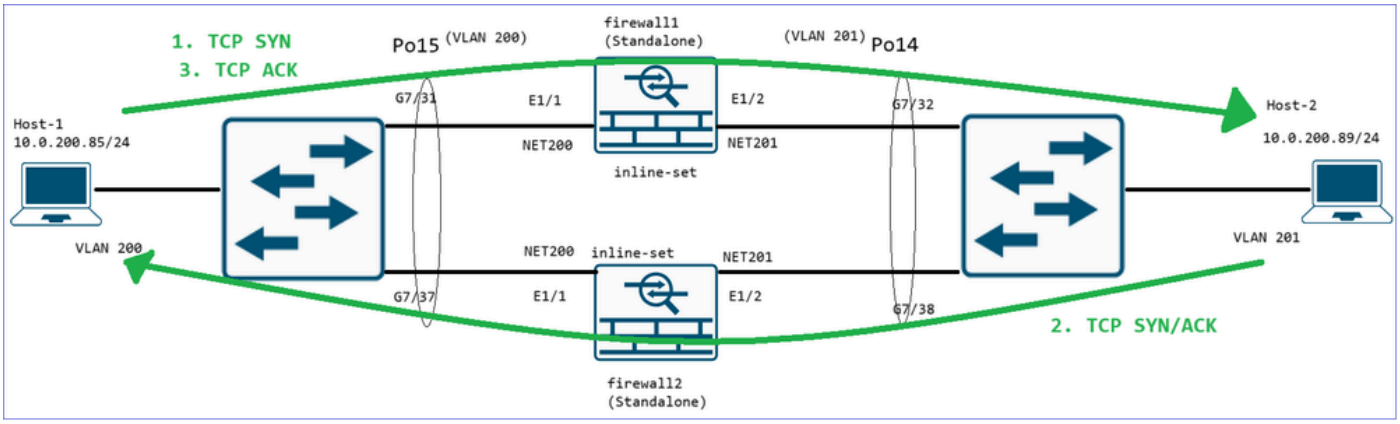
Warning: On inline-set (IPS-mode) interfaces, TCP State Bypass is enabled by default. If TSID or a TLS decryption policy is configured in ACP Advanced Settings, it silently overrides/disables this default TCP State Bypass behavior, so TCP state checks are enforced instead of being bypassed. Observable symptom: For TCP state bypass-eligible traffic traversing an inline set (e.g., asymmetric traffic scenarios where traffic egresses and ingresses different inline sets), **packets can be dropped with ASP drop tcp-not-syn** (syslog 106015).

Consider this scenario:



The two firewalls run in **standalone** mode (they even run different software versions), but are handling traffic from the same port-channel interfaces.

In this case, you can have asymmetric traffic due to port-channel load-balancing algorithm:



1. Host-1 (10.0.200.85) sends a TCP **SYN** towards Host-2 (10.0.200.89). This packet traverses **firewall1.**
2. Host-2 (10.0.200.89) sends a TCP **SYN/ACK** towards Host-2 (10.0.200.85). This packet traverses **firewall2.**
3. The TCP 3-way handshake is completed with Host-1 sending a TCP **ACK** towards Host-2. This packet traverses **firewall1**.

From Host-1 point of view, the connection is successfully established:

<#root>

```
root@kali:/ #
```

wget -O - http://10.0.200.89/10K

```
--2025-05-06 08:20:28--  http://10.0.200.89/10K
Connecting to 10.0.200.89:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 10240 (10K)
Saving to: ‘STDOUT’
-                                           100%[=======================================================
2025-05-06 08:20:28 (99.8 MB/s) - written to stdout [10240/10240]
```

Packet capture on firewall1 shows only traffic from Host1 to Host2:

<#root>

```
firepower#
```

show capture

```
capture CAPI type raw-data trace interface NET200 [Capturing - 875 bytes]
  match ip host 10.0.200.85 host 10.0.200.89
capture CAPO type raw-data trace interface NET201 [Capturing - 875 bytes]
  match ip host 10.0.200.85 host 10.0.200.89
```

The capture contents:

<#root>

```
firepower#
```

show capture CAPI

```
9 packets captured
   1: 12:21:14.161689       10.0.200.85.44806 > 10.0.200.89.80:
```

S

```
1877376557:1877376557(0) win 64240 <mss 1460,sackOK,timestamp 2133104674 0,nop,wscale 7>
   2: 12:21:14.162924       10.0.200.85.44806 > 10.0.200.89.80: .
```

ack

```
 3274105192 win 502 <nop,nop,timestamp 2133104676 1658009126>
   3: 12:21:14.163077       10.0.200.85.44806 > 10.0.200.89.80: P 1877376558:1877376687(129) ack 327410
   4: 12:21:14.164801       10.0.200.85.44806 > 10.0.200.89.80: . ack 3274106640 win 501 <nop,nop,times
   5: 12:21:14.164908       10.0.200.85.44806 > 10.0.200.89.80: . ack 3274108088 win 494
...
```

Packet capture on firewall2 shows only traffic from Host2 to Host1:

<#root>

FTD1010-12#

show capture CAPI

```
11 packets captured
   1: 12:21:14.198949       10.0.200.89.80 > 10.0.200.85.44806:
```

S

```
 3274105191:3274105191(0)
```

ack

```
 1877376558 win 65160 <mss 1460,sackOK,timestamp 1658009126 2133104674,nop,wscale 7>
   2: 12:21:14.200001       10.0.200.89.80 > 10.0.200.85.44806: . ack 1877376687 win 509 <nop,nop,times
   3: 12:21:14.200825       10.0.200.89.80 > 10.0.200.85.44806: . 3274105192:3274106640(1448) ack 18773
   4: 12:21:14.200947       10.0.200.89.80 > 10.0.200.85.44806: . 3274106640:3274108088(1448) ack 18773
   5: 12:21:14.200963       10.0.200.89.80 > 10.0.200.85.44806: . 3274108088:3274109536(1448) ack 18773
   6: 12:21:14.200978       10.0.200.89.80 > 10.0.200.85.44806: . 3274109536:3274110984(1448) ack 18773
   7: 12:21:14.200993       10.0.200.89.80 > 10.0.200.85.44806: P 3274110984:3274112432(1448) ack
...
```

Syslogs on firewall1 show that the TCP SYN packet created a TCP state-bypass connection:

<#root>

firepower#

show logging

```
...
May 06 2025 12:21:14: %FTD-6-302303:
```

```
Built TCP state-bypass connection

 106977 from NET200:10.0.200.85/44806 (10.0.200.85/44806) to NET201:10.0.200.89/80 (10.0.200.89/80)
```

On firewall2, the TCP SYN/ACK packet also created a TCP state-bypass connection:

```
<#root>

FTD1010-12#

show logging


...
May 06 2025 12:21:14: %FTD-6-302303:

Built TCP state-bypass


connection

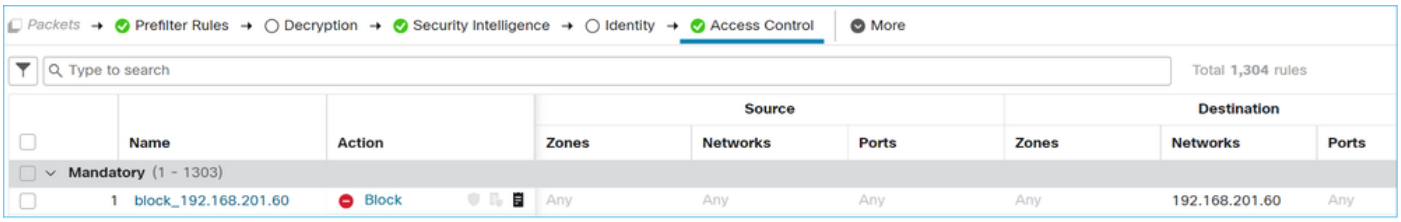
 325 from NET201:10.0.200.89/80 (10.0.200.89/80) to NET200:10.0.200.85/44806 (10.0.200.85/44806)
```

Main Points

- Asymmetric traffic is working through an inline set of different firewall devices since both devices handle the TCP connection in TCP state-bypass mode.
- Note that TCP state-bypass is not manually configured on the firewalls, but it is the result of the inline-set interface operation.

Block Packet on Inline Pair Interface Mode

Create a Block rule, send traffic through the FTD Inline Pair, and observe the behavior, as shown in the image.



Solution

Enable capture with trace and send the SYN/ACK packets through the FTD Inline Pair. The traffic is blocked:

<#root>

firepower#

show capture

capture CAPI type raw-data trace interface INSIDE

[Capturing - 270 bytes]

  match ip host 192.168.201.60 any
capture CAPO type raw-data interface OUTSIDE

[Capturing - 0 bytes]

  match ip host 192.168.201.60 any

In the trace, it can be seen that the packet was dropped by the FTD LINA engine and was not forwarded to the FTD Snort engine.

<#root>

firepower#

show capture CAPI packet-number 1 trace

4 packets captured

   1: 09:41:54.562547   192.168.201.50.59144 > 192.168.201.60.80: S 3817586151:3817586151(0) win 64
Phase: 1
Type: NGIPS-MODE
Subtype: ngips-mode

```
Result: ALLOW
Elapsed time: 10126 ns
Config:
Additional Information:
The flow ingressed an interface configured for NGIPS mode and NGIPS services will be applied

Phase: 2
Type: ACCESS-LIST
Subtype:
Result: DROP
Elapsed time: 10126 ns
Config:
access-group CSM_FW_ACL_ global
access-list CSM_FW_ACL_ advanced deny ip any host 192.168.201.60 rule-id 268451045 event-log flow-start
access-list CSM_FW_ACL_ remark rule-id 268451045: ACCESS POLICY: mzafeiro_2m - Mandatory
access-list CSM_FW_ACL_ remark rule-id 268451045: L4 RULE: block_192.168.201.60
Additional Information:

Result:
input-interface: INSIDE(vrfid:0)
input-status: up
input-line-status: up
Action: drop

Time Taken: 20252 ns

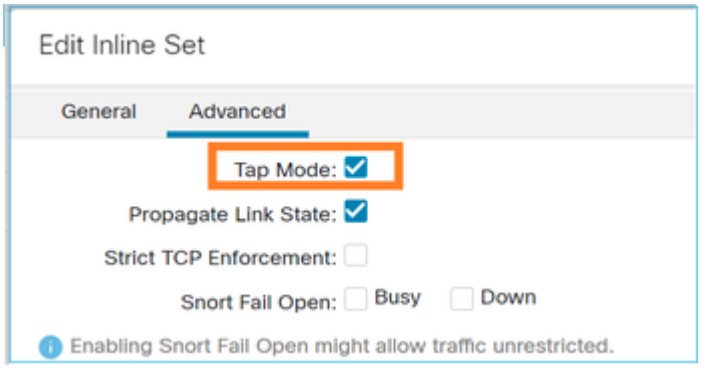
1 packet shown
```

Configure Inline Pair Mode With Tap

Enable Tap mode on the Inline Pair.

Solution

Navigate to **Devices > Device Management > Inline Sets > Edit Inline Set > Advanced** and enable **Tap Mode**, as shown in the image.



Verification

<#root>

```
firepower#
```

```
 show inline-set
```

```
Inline-set Inline-Pair-1
  Mtu is 1500 bytes
  Fail-open for snort down is off
  Fail-open for snort busy is off
```

```
Tap mode is on
```

```
  Propagate-link-state option is on
  hardware-bypass mode is disabled
  Interface-Pair[1]:
    Interface: Ethernet1/4 "OUTSIDE"
      Current-Status: UP
    Interface: Ethernet1/3 "INSIDE"
      Current-Status: UP
```

```
   Bridge Group ID: 0
```

Verify FTD Inline Pair With Tap Interface Operation

Basic Theory

- When you configure an Inline Pair with Tap, physical interfaces are internally bridged.
- It is available in Routed or Transparent Deployment modes.
- Most of LINA engine features (NAT, Routing, and so on) are not available for flows which go through the Inline Pair.
- Actual traffic cannot be dropped.
- A few LINA engine checks are applied along with full Snort engine checks to a copy of the actual traffic.

Inline Pair with Tap Mode does not drop the transit traffic. With the trace of a packet it confirms this:

<#root>

```
>
```

show capture CAPI packet-number 2 trace

```
3 packets captured

   2: 13:34:30.685084       192.168.201.50.20 > 192.168.201.60.80: S 0:0(0) win 8192
Phase: 1
Type: CAPTURE
Subtype:
Result: ALLOW
Config:
Additional Information:
MAC Access list

Phase: 2
Type: ACCESS-LIST
Subtype:
Result: ALLOW
Config:
Implicit Rule
Additional Information:
MAC Access list
```

Phase: 3
Type: NGIPS-MODE
Subtype: ngips-mode

```
Result: ALLOW
Config:
Additional Information:
```

The flow ingressed an interface configured for NGIPS mode and NGIPS services is applied

Phase: 4
Type: ACCESS-LIST
Subtype: log
Result: WOULD HAVE DROPPED

```
Config:
access-group CSM_FW_ACL_ global
access-list CSM_FW_ACL_ advanced deny ip 192.168.201.0 255.255.255.0 any rule-id 268441600 event-log fl
access-list CSM_FW_ACL_ remark rule-id 268441600: ACCESS POLICY: FTD4100 - Mandatory/1
access-list CSM_FW_ACL_ remark rule-id 268441600: L4 RULE: Rule 1
Additional Information:

Result:
input-interface: INSIDE
input-status: up
input-line-status: up
```

Action: Access-list would have dropped, but packet forwarded due to inline-tap

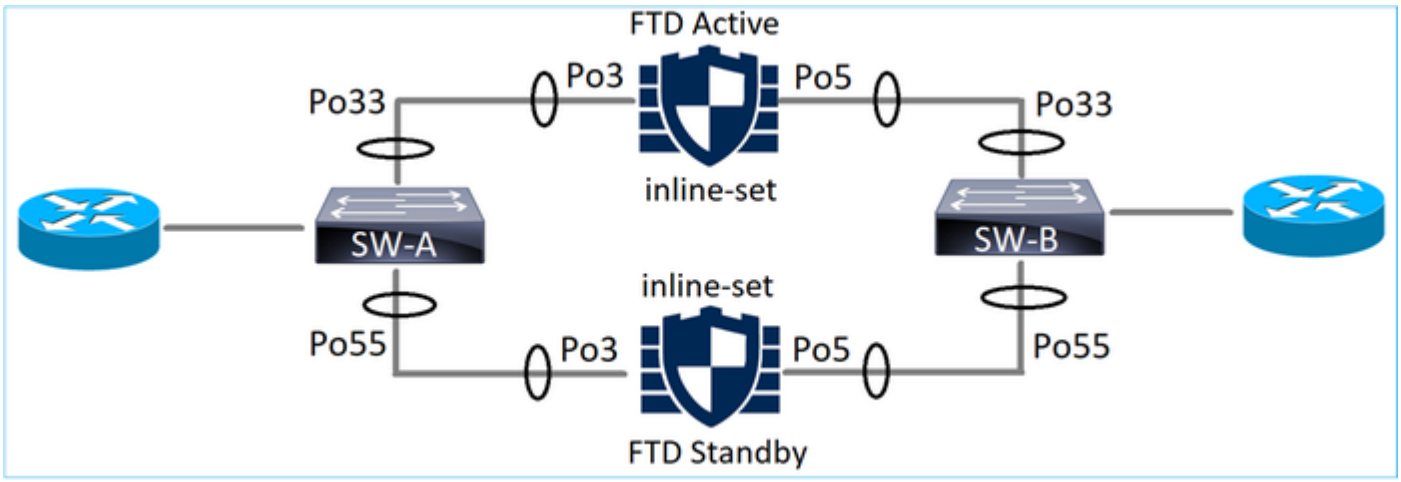
```
1 packet shown
>
```

Inline Pair and Etherchannel

You can configure inline pair with etherchannel in 2 ways:

1. Etherchannel terminated on FTD.
2. Etherchannel goes through the FTD (requires FXOS code 2.3.1.3 and later).

Etherchannel Terminated on FTD



Etherchannels on SW-A:

<#root>

SW-A#

show etherchannel summary | i Po33|Po55

```
33     Po33(SU)         LACP      Gi3/11(P)
35     Po55(SU)         LACP      Gi2/33(P)
```

Etherchannels on SW-B:

<#root>

SW-B#

```
show etherchannel summary | i Po33|Po55

33     Po33(SU)        LACP        Gi1/0/3(P)
55     Po55(SU)        LACP        Gi1/0/4(P)
```

The traffic is forwarded through the Active FTD based on MAC address learning:

```
<#root>

SW-B#

 show mac address-table address 0017.dfd6.ec00

          Mac Address Table
------------------------------------------

Vlan    Mac Address       Type        Ports
----    -----------       --------    -----
 201    0017.dfd6.ec00    DYNAMIC

Po33

Total Mac Addresses for this criterion: 1
```

The inline-set on FTD:

```
<#root>

FTD#

show inline-set

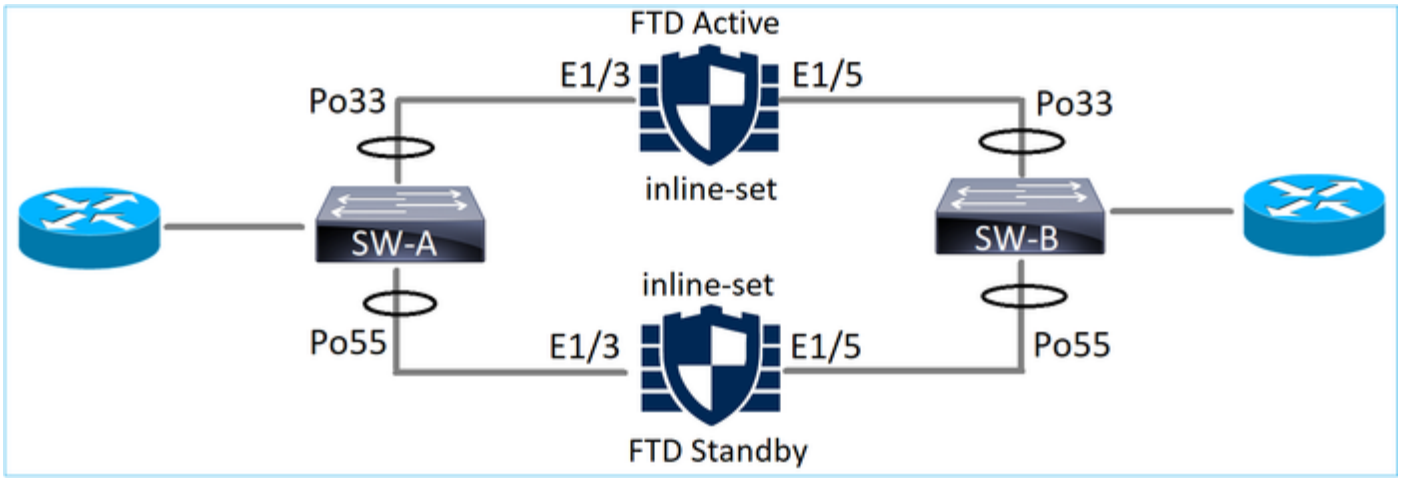

Inline-set SET1
  Mtu is 1500 bytes
  Fail-open for snort down is on
  Fail-open for snort busy is off
  Tap mode is off
  Propagate-link-state option is off
  hardware-bypass mode is disabled


Interface-Pair[1]:
    Interface: Port-channel3 "INSIDE"
      Current-Status: UP
    Interface: Port-channel5 "OUTSIDE"
      Current-Status: UP

    Bridge Group ID: 775
```

Note: In case of an FTD failover event, the traffic outage depends mainly on the time it takes the switches to learn the MAC address of the remote peer.

Etherchannel through the FTD



Etherchannels on SW-A:

<#root>

SW-A#

show etherchannel summary | i Po33|Po55

```
33     Po33(SU)         LACP      Gi3/11(P)
55     Po55(SD)         LACP      Gi3/7
```

(I)

The LACP packets through the Standby FTD are blocked:

<#root>

FTD#

capture ASP type asp-drop fo-standby

FTD#

show capture ASP | i 0180.c200.0002

```
  29: 15:28:32.658123       a0f8.4991.ba03 0180.c200.0002 0x8809 Length: 124
```

```
  70: 15:28:47.248262       f0f7.556a.11e2 0180.c200.0002 0x8809 Length: 124
```

Etherchannels on SW-B:

<#root>

```
SW-B#
```

show etherchannel summary | i Po33|Po55

```
33     Po33(SU)        LACP        Gi1/0/3(P)
55     Po55(SD)        LACP        Gi1/0/4
```

(s)

The traffic is forwarded through the Active FTD based on MAC address learning:

<#root>

```
SW-B#
```

show mac address-table address 0017.dfd6.ec00

```
          Mac Address Table
-------------------------------------------

Vlan    Mac Address       Type        Ports
----    -----------       --------    -----
 201    0017.dfd6.ec00    DYNAMIC
```

Po33

```
Total Mac Addresses for this criterion: 1
```

The inline-set on FTD:

<#root>

```
FTD#
```

show inline-set

```
Inline-set SET1
  Mtu is 1500 bytes
  Fail-open for snort down is on
  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 "INSIDE"


      Current-Status: UP


    Interface: Ethernet1/5 "OUTSIDE"


      Current-Status: UP


    Bridge Group ID: 519
```

⚠ **Caution**: In this scenario, in case of an FTD failover event, the convergence time mainly depends on the Etherchannel LACP negotiation, and on the time it takes the outage, which can be quite longer. In case the Etherchannel mode is ON (no LACP), then the convergence time depends on MAC address learning.

Troubleshoot

There is currently no specific information available for this configuration.

Comparison: Inline Pair vs Inline Pair with Tap

	Inline Pair	**Inline Pair with Tap**
show inline-	> show inline-set	> show inline-set

set	Inline-set Inline-Pair-1
	  Mtu is 1500 bytes
	  Failsafe mode is on/activated
	  Failsecure mode is off
	  Tap mode is off
	  Propagate-link-state option is on
	  hardware-bypass mode is disabled
	  Interface-Pair[1]:
	  Interface: Ethernet1/6 "INSIDE"
	  Current-Status: UP
	  Interface: Ethernet1/8 "OUTSIDE"
	  Current-Status: UP
	  Bridge Group ID: 509
	>
		Inline-set Inline-Pair-1
		  Mtu is 1500 bytes
		  Failsafe mode is on/activated
		  Failsecure mode is off
		  Tap mode is on
		  Propagate-link-state option is on
		  hardware-bypass mode is disabled
		  Interface-Pair[1]:
		  Interface: Ethernet1/6 "INSIDE"
		  Current-Status: UP
		  Interface: Ethernet1/8 "OUTSIDE"
		  Current-Status: UP
		  Bridge Group ID: 0
		
		>

show interface	> show interface e1/6
	Interface Ethernet1/6 "INSIDE", is up, line protocol is up
	  Hardware is EtherSVI, BW 1000 Mbps, DLY 1000 usec
	  MAC address 5897.bdb9.770e, MTU 1500
	  IPS Interface-Mode: inline, Inline-Set: Inline-Pair-1
	  IP address unassigned
	  Traffic Statistics for "INSIDE":
	  3957 packets input, 264913 bytes
	  144 packets output, 58664 bytes
	  4 packets dropped
	  1 minute input rate 0 pkts/sec, 26 bytes/sec
	  1 minute output rate 0 pkts/sec, 7 bytes/sec
	  1 minute drop rate, 0 pkts/sec
	  5 minute input rate 0 pkts/sec, 28 bytes/sec
	  5 minute output rate 0 pkts/sec, 9 bytes/sec
	  5 minute drop rate, 0 pkts/sec
	> show interface e1/8
	Interface Ethernet1/8 "OUTSIDE", is up, line protocol is up
	  Hardware is EtherSVI, BW 1000 Mbps, DLY 1000 usec
	  MAC address 5897.bdb9.774d, MTU 1500
	  IPS Interface-Mode: inline, Inline-Set: Inline-Pair-1
	  IP address unassigned
	  Traffic Statistics for "OUTSIDE":
	  144 packets input, 55634 bytes
		> show interface e1/6
		Interface Ethernet1/6 "INSIDE", is up, line protocol is up
		  Hardware is EtherSVI, BW 1000 Mbps, DLY 1000 usec
		  MAC address 5897.bdb9.770e, MTU 1500
		  IPS Interface-Mode: inline-tap, Inline-Set: Inline-Pair-1
		  IP address unassigned
		  Traffic Statistics for "INSIDE":
		  24 packets input, 1378 bytes
		  0 packets output, 0 bytes
		  24 packets dropped
		  1 minute input rate 0 pkts/sec, 0 bytes/sec
		  1 minute output rate 0 pkts/sec, 0 bytes/sec
		  1 minute drop rate, 0 pkts/sec
		  5 minute input rate 0 pkts/sec, 0 bytes/sec
		  5 minute output rate 0 pkts/sec, 0 bytes/sec
		  5 minute drop rate, 0 pkts/sec
		> show interface e1/8
		Interface Ethernet1/8 "OUTSIDE", is up, line protocol is up
		  Hardware is EtherSVI, BW 1000 Mbps, DLY 1000 usec
		  MAC address 5897.bdb9.774d, MTU 1500
		  IPS Interface-Mode: inline-tap, Inline-Set: Inline-Pair-1
		  IP address unassigned
		  Traffic Statistics for "OUTSIDE":
		  1 packets input, 441 bytes
		  0 packets output, 0 bytes
		  1 packets dropped

	3954 packets output, 339987 bytes 0 packets dropped 1 minute input rate 0 pkts/sec, 7 bytes/sec 1 minute output rate 0 pkts/sec, 37 bytes/sec 1 minute drop rate, 0 pkts/sec 5 minute input rate 0 pkts/sec, 8 bytes/sec 5 minute output rate 0 pkts/sec, 39 bytes/sec 5 minute drop rate, 0 pkts/sec >	1 minute input rate 0 pkts/sec, 0 bytes/sec 1 minute output rate 0 pkts/sec, 0 bytes/sec 1 minute drop rate, 0 pkts/sec 5 minute input rate 0 pkts/sec, 0 bytes/sec 5 minute output rate 0 pkts/sec, 0 bytes/sec 5 minute drop rate, 0 pkts/sec >
To Handle Packet with Block rule	> show capture CAPI packet-number 1 trace 3 packets captured 1: 16:12:55.785085 192.168.201.50.20 > 192.168.201.60.80: S 0:0(0) ack 0 win 8192 Phase: 1 Type: CAPTURE Subtype: Result: ALLOW Config: Additional Information: MAC Access list Phase: 2 Type: ACCESS-LIST Subtype: Result: ALLOW Config: Implicit Rule Additional Information: MAC Access list Phase: 3 Type: NGIPS-MODE Subtype: ngips-mode Result: ALLOW Config: Additional Information: The flow ingressed an interface configured for NGIPS mode and NGIPS services is applied Phase: 4 Type: ACCESS-LIST Subtype: log Result: DROP Config: access-group CSM_FW_ACL_ global access-list CSM_FW_ACL_ advanced deny	> show capture CAPI packet-number 1 trace 3 packets captured 1: 16:56:02.631437 192.168.201.50.20 > 192.168.201.60.80: S 0:0(0) win 8192 Phase: 1 Type: CAPTURE Subtype: Result: ALLOW Config: Additional Information: MAC Access list Phase: 2 Type: ACCESS-LIST Subtype: Result: ALLOW Config: Implicit Rule Additional Information: MAC Access list Phase: 3 Type: NGIPS-MODE Subtype: ngips-mode Result: ALLOW Config: Additional Information: The flow ingressed an interface configured for NGIPS mode and NGIPS services is applied Phase: 4 Type: ACCESS-LIST Subtype: log Result: WOULD HAVE DROPPED Config: access-group CSM_FW_ACL_ global access-list CSM_FW_ACL_ advanced deny ip 192.168.201.0 255.255.255.0 any rule-id

	ip 192.168.201.0 255.255.255.0 any rule-id 268441600 event-log flow-start access-list CSM_FW_ACL_ remark rule-id 268441600: ACCESS POLICY: FTD4100 - Mandatory/1 access-list CSM_FW_ACL_ remark rule-id 268441600: L4 RULE: Rule 1 Additional Information: Result: input-interface: INSIDE input-status: up input-line-status: up Action: drop Drop-reason: (acl-drop) Flow is denied by configured rule 1 packet shown >	268441600 event-log flow-start access-list CSM_FW_ACL_ remark rule-id 268441600: ACCESS POLICY: FTD4100 - Mandatory/1 access-list CSM_FW_ACL_ remark rule-id 268441600: L4 RULE: Rule 1 Additional Information: Result: input-interface: INSIDE input-status: up input-line-status: up Action: Access-list would have dropped,but packet forwarded due to inline-tap 1 packet shown >

Summary

- When you use Inline Pair mode, the packet goes mainly through the FTD Snort engine.
- TCP connections are handled in a TCP state-bypass mode.
- From an FTD LINA engine point of view, an ACL policy is applied.
- When Inline Pair Mode is in use, packets can be blocked since they are processed inline.
- When Tap Mode is enabled, a copy of the packet is inspected and dropped internally while the actual traffic goes through FTD unmodified.

Related Information

- Cisco Firepower NGFW
- Technical Support & Documentation - Cisco Systems