

Troubleshoot Firepower Threat Defense Routing

Contents

[Introduction](#)

[Prerequisites](#)

[Requirements](#)

[Components Used](#)

[Background Information](#)

[FTD Packet Forwarding Mechanisms](#)

[Key Point](#)

[Data-plane \(LINA\) Routing Behavior](#)

[Key Points](#)

[FTD Order of Operations](#)

[Configure](#)

[Case 1 – “Forwarding Based on Connection Lookup](#)

[Floating Timeout](#)

[Conn-holddown Timeout](#)

[Case 2 – “Forwarding Based on NAT Lookup](#)

[Case 3 – “Forwarding based on Policy Based Routing \(PBR\)](#)

[Case 4 – “Forwarding based on Global Routing Lookup](#)

[Null0 Interface](#)

[Equal Cost Multi-Path \(ECMP\)](#)

[FTD Management Plane](#)

[FTD LINA Diagnostic Interface Routing](#)

Introduction

This document describes how Firepower Threat Defense (FTD) forwards packets and implements various routing concepts.

Prerequisites

Requirements

- Basic routing knowledge

Components Used

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

- Cisco Firepower 41xx Threat Defense Version 7.1.x
- Firepower Management Center (FMC) Version 7.1.x

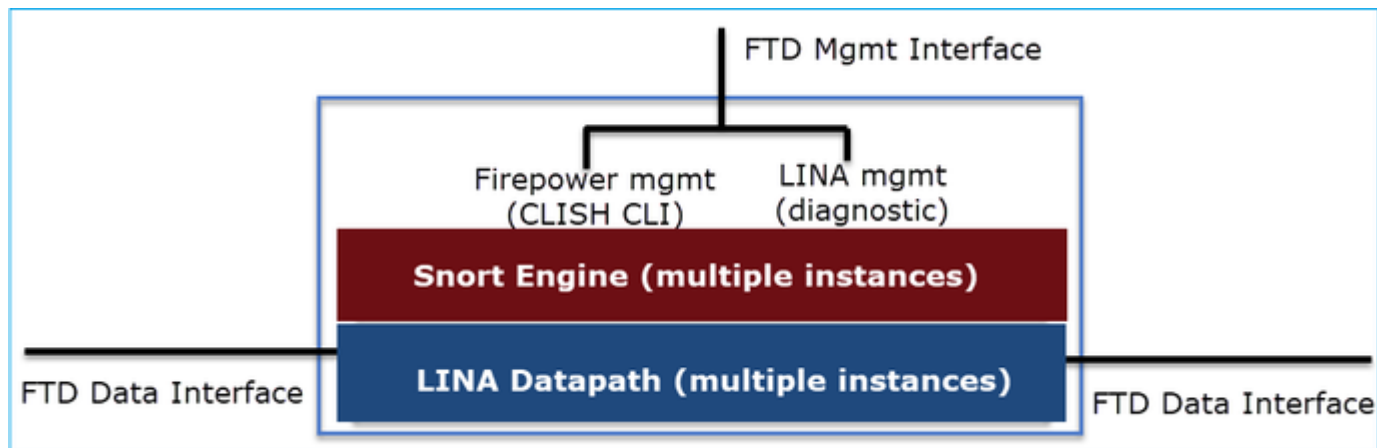
The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, ensure that you understand the potential impact of any command.

Background Information

FTD Packet Forwarding Mechanisms

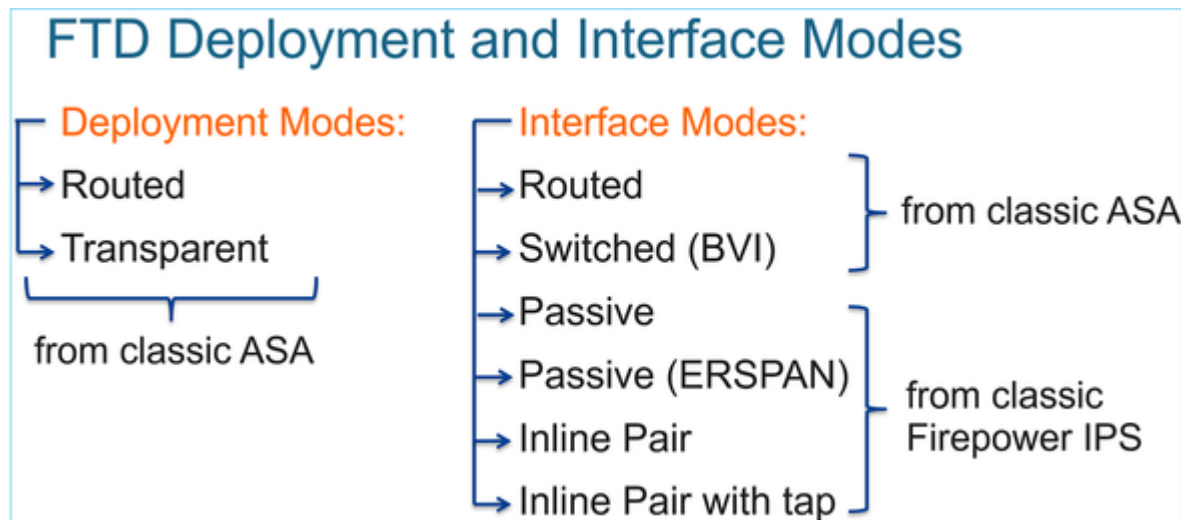
FTD is a unified software image that consists of 2 main engines:

- Datapath engine (LINA)
- Snort engine



The Datapath and the Snort Engine are the main parts of the FTD's Data Plane.

The FTD Data Plane forwarding mechanism depends on the interface mode. The next picture summarizes the various interface modes along with the FTD deployment modes:



The table summarizes how the FTD forwards packets in the data plane based on the interface mode. The forwarding mechanisms are listed in order of preference:

FTD Deployment mode	FTD Interface mode	Forwarding Mechanism
Routed	Routed	Packet forwarding based on the following order: 1. Connection lookup 2. Nat lookup (xlate) 3. Policy Based Routing (PBR) 4. Global routing table lookup
Routed or Transparent	Switched (BVI)	1. NAT lookup 2. Destination MAC Address L2 Lookup*
Routed or Transparent	Inline Pair	The packet will be forwarded based on the pair configuration.
Routed or Transparent	Inline Pair with Tap	The original packet will be forwarded based on the pair configuration. The copy of the packet will be dropped internally
Routed or Transparent	Passive	The packet is dropped internally
Routed	Passive (ERSPAN)	The packet is dropped internally

* An FTD in Transparent mode does a Route Lookup in some situations:

MAC Address vs. Route Lookups

For traffic within a bridge group, the outgoing interface of a packet is determined by performing a destination MAC address lookup instead of a route lookup.

Route lookups, however, are necessary for the following situations:

- Traffic originating on the Firepower Threat Defense device—Add a default/static route on the Firepower Threat Defense device for traffic destined for a remote network where a syslog server, for example, is located.
- Voice over IP (VoIP) and TFTP traffic, and the endpoint is at least one hop away—Add a static route on the Firepower Threat Defense device for traffic destined for the remote endpoint so that secondary connections are successful. The Firepower Threat Defense device creates a temporary "pinhole" in the access control policy to allow the secondary connection; and because the connection might use a different set of IP addresses than the primary connection, the Firepower Threat Defense device needs to perform a route lookup to install the pinhole on the correct interface.

Affected applications include:

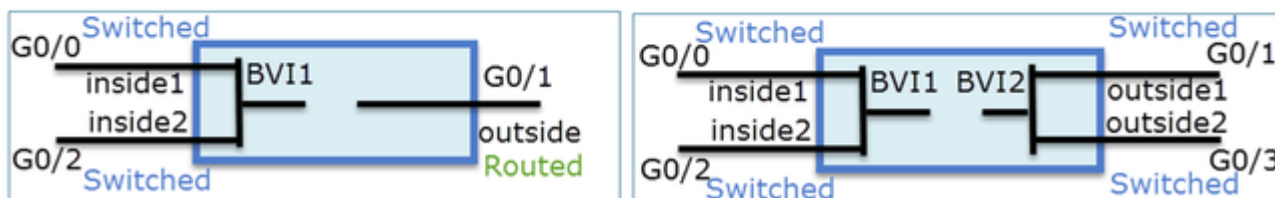
- H.323
- RTSP
- SIP
- Skinny (SCCP)
- SQL*Net
- SunRPC
- TFTP
- Traffic at least one hop away for which the Firepower Threat Defense device performs NAT—Configure a static route on the Firepower Threat Defense device for traffic destined for the remote network. You also need a static route on the upstream router for traffic destined for the mapped addresses to be sent to the Firepower Threat Defense device.

Check the [FMC guide](#) for more details.

As from the 6.2.x version, the FTD supports Integrated Routing and Bridging (IRB):

FTD Integrated Routing and Bridging (IRB)

- Available as from 6.2.x
- Allows an FTD in **Routed mode** to have multiple interfaces (up to 64) to be part of the **same VLAN** and perform L2 switching between them
- BVI-to-Routed or BVI-to-BVI Routing is allowed



BVI verification commands:

Verification commands

```
firepower# show bridge-group
```

```
firepower# show ip
```

Interface	Name	IP address	Subnet mask	Method
GigabitEthernet0/0	VLAN1576_G0-0	203.0.113.1	255.255.255.0	manual
GigabitEthernet0/1	VLAN1577_G0-1	192.168.1.15	255.255.255.0	manual
GigabitEthernet0/2	VLAN1576_G0-2	203.0.113.1	255.255.255.0	manual
GigabitEthernet0/4.100	SUB1	203.0.113.1	255.255.255.0	manual
BVI1	LAN	203.0.113.1	255.255.255.0	manual
BVI2	LAN2	192.168.1.15	255.255.255.0	manual

- BVI nameif is used in L3 Routing configuration

```
firepower# show run route
```

```
route LAN 1.1.1.0 255.255.255.0 203.0.113.5 1
```

- BVI member nameif is used in policies like NAT configuration

```
firepower# show run nat
```

```
nat (VLAN1576_G0-0,VLAN1577_G0-1) source dynamic any interface  
nat (VLAN1576_G0-2,VLAN1577_G0-1) source dynamic any interface
```

Key Point

For routed interfaces or BVIs (IRB) the packet forwarding is based on this order:

- Connection lookup
- NAT lookup (destination NAT, also known as UN-NAT)
- Policy-Based Routing (PBR)
- Global routing table lookup

What about source NAT?

The source NAT is checked after the Global routing lookup.

The rest of this document focuses on the Routed interface mode.

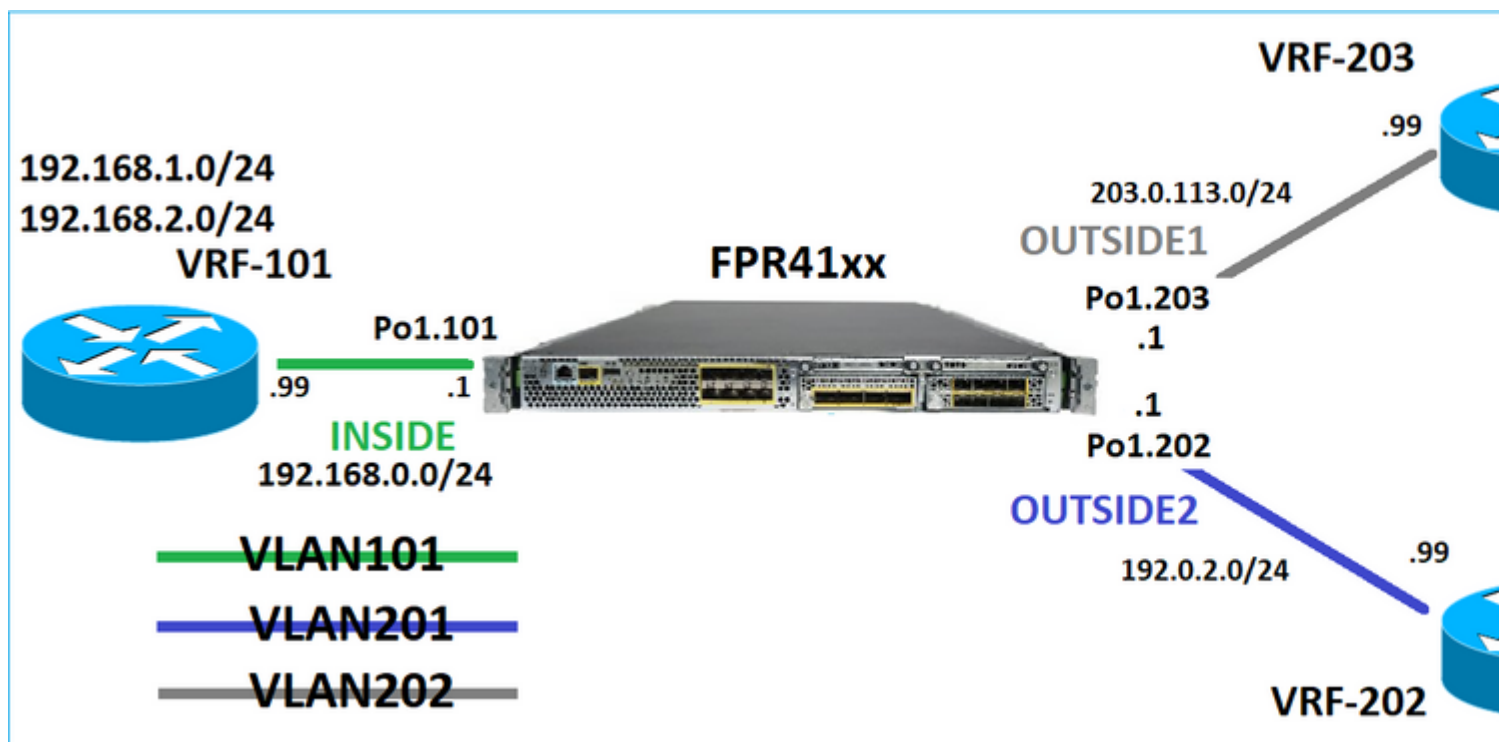
Data-plane (LINA) Routing Behavior

In routed interface mode FTD LINA forwards the packets in 2 phases:

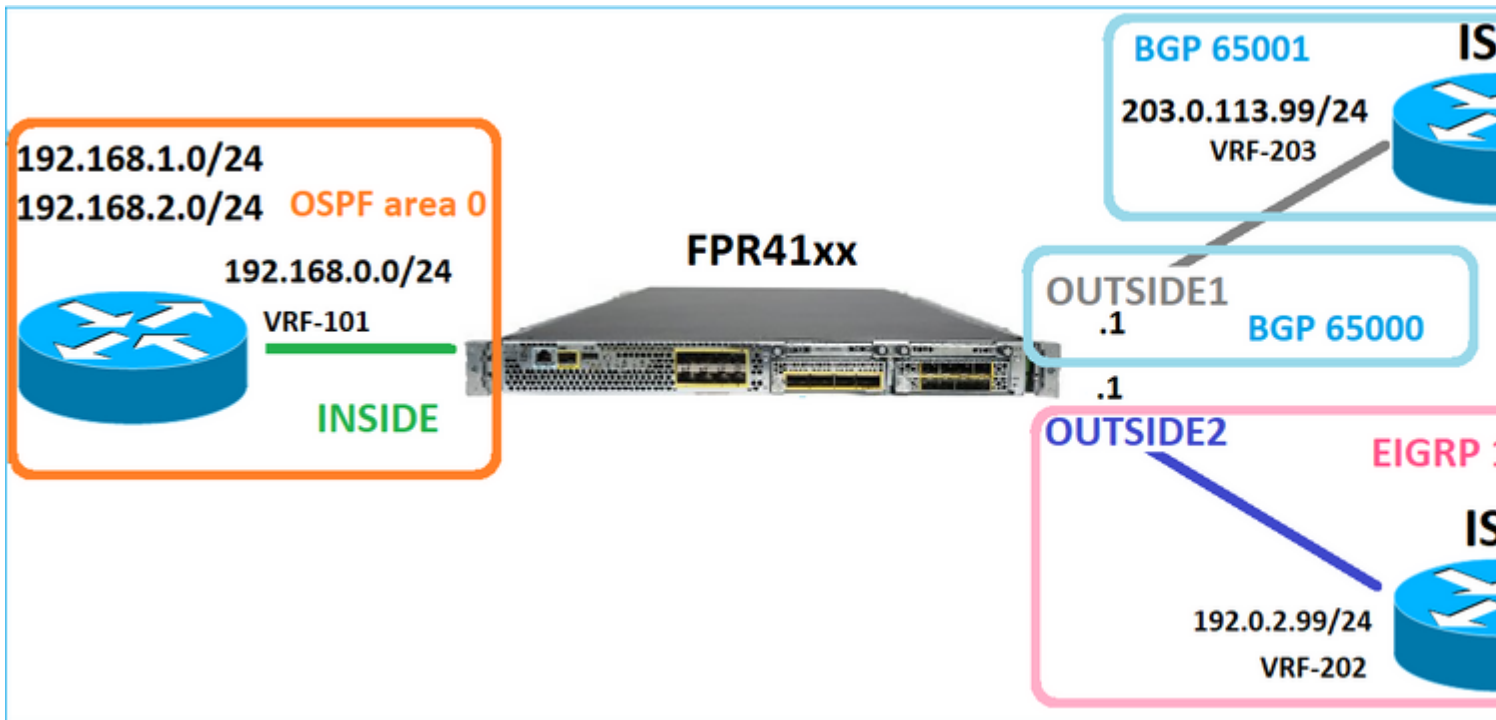
Phase 1 – Egress Interface Determination

Phase 2 – Next-Hop Selection

Consider this topology:



And this routing design:



The FTD routing configuration:

```
firepower# show run router
router ospf 1
network 192.168.0.0 255.255.255.0 area 0
log-adj-changes
!
router bgp 65000
bgp log-neighbor-changes
bgp router-id vrf auto-assign
address-family ipv4 unicast
neighbor 203.0.113.99 remote-as 65001
neighbor 203.0.113.99 ebgp-multihop 255
neighbor 203.0.113.99 transport path-mtu-discovery disable
neighbor 203.0.113.99 activate
no auto-summary
no synchronization
exit-address-family
!
router eigrp 1
no default-information in
no default-information out
no eigrp log-neighbor-warnings
no eigrp log-neighbor-changes
network 192.0.2.0 255.255.255.0
!
firepower# show run route
route OUTSIDE2 198.51.100.0 255.255.255.248 192.0.2.99 1
```

The FTD Routing Information Base (RIB) - Control Plane:

```
firepower# show route | begin Gate
```

Gateway of last resort is not set

```
C 192.0.2.0 255.255.255.0 is directly connected, OUTSIDE2
L 192.0.2.1 255.255.255.255 is directly connected, OUTSIDE2
C 192.168.0.0 255.255.255.0 is directly connected, INSIDE
L 192.168.0.1 255.255.255.255 is directly connected, INSIDE
O 192.168.1.1 255.255.255.255
[110/11] via 192.168.0.99, 01:11:25, INSIDE
O 192.168.2.1 255.255.255.255
[110/11] via 192.168.0.99, 01:11:15, INSIDE
S 198.51.100.0 255.255.255.248 [1/0] via 192.0.2.99, OUTSIDE2
D 198.51.100.8 255.255.255.248
[90/130816] via 192.0.2.99, 01:08:11, OUTSIDE2
D 198.51.100.16 255.255.255.248
[90/130816] via 192.0.2.99, 01:08:04, OUTSIDE2
B 198.51.100.24 255.255.255.248 [20/0] via 203.0.113.99, 00:28:29
B 198.51.100.32 255.255.255.248 [20/0] via 203.0.113.99, 00:28:16
C 203.0.113.0 255.255.255.0 is directly connected, OUTSIDE1
L 203.0.113.1 255.255.255.255 is directly connected, OUTSIDE1
```

The corresponding FTD Accelerated Security Path (ASP) Routing table - Data Plane:

```
firepower# show asp table routing
route table timestamp: 91
in 169.254.1.1 255.255.255.255 identity
in 192.168.0.1 255.255.255.255 identity
in 192.0.2.1 255.255.255.255 identity
in 192.168.1.1 255.255.255.255 via 192.168.0.99, INSIDE
in 192.168.2.1 255.255.255.255 via 192.168.0.99, INSIDE
in 203.0.113.1 255.255.255.255 identity
in 169.254.1.0 255.255.255.248 nlp_int_tap
in 198.51.100.0 255.255.255.248 via 192.0.2.99, OUTSIDE2
in 198.51.100.8 255.255.255.248 via 192.0.2.99, OUTSIDE2
in 198.51.100.16 255.255.255.248 via 192.0.2.99, OUTSIDE2
in 198.51.100.24 255.255.255.248 via 203.0.113.99 (unresolved, timestamp: 89)
in 198.51.100.32 255.255.255.248 via 203.0.113.99 (unresolved, timestamp: 90)
in 192.168.0.0 255.255.255.0 INSIDE
in 192.0.2.0 255.255.255.0 OUTSIDE2
in 203.0.113.0 255.255.255.0 OUTSIDE1
in ff02::1 ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff identity
in ff02::1:ff01:3 ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff identity
in ff02::1:ff00:1 ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff identity
in fe80::200:ff:fe01:3 ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff identity
in fd00:0:0:1::1 ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff identity
in fd00:0:0:1:: ffff:ffff:ffff:ffff:: nlp_int_tap
out 255.255.255.255 255.255.255.255 OUTSIDE1
out 203.0.113.1 255.255.255.255 OUTSIDE1
out 203.0.113.0 255.255.255.0 OUTSIDE1
out 224.0.0.0 240.0.0.0 OUTSIDE1
out 255.255.255.255 255.255.255.255 OUTSIDE2
out 192.0.2.1 255.255.255.255 OUTSIDE2
out 198.51.100.0 255.255.255.248 via 192.0.2.99, OUTSIDE2
out 198.51.100.8 255.255.255.248 via 192.0.2.99, OUTSIDE2
out 198.51.100.16 255.255.255.248 via 192.0.2.99, OUTSIDE2
out 192.0.2.0 255.255.255.0 OUTSIDE2
out 224.0.0.0 240.0.0.0 OUTSIDE2
out 255.255.255.255 255.255.255.255 INSIDE
out 192.168.0.1 255.255.255.255 INSIDE
```



```

out 192.168.1.1 255.255.255.255 via 192.168.0.99, INSIDE
out 192.168.2.1 255.255.255.255 via 192.168.0.99, INSIDE
out 192.168.0.0 255.255.255.0 INSIDE
out 224.0.0.0 240.0.0.0 INSIDE
out 255.255.255.255 255.255.255.255 cmi_mgmt_int_tap
out 224.0.0.0 240.0.0.0 cmi_mgmt_int_tap
out 255.255.255.255 255.255.255.255 ha_ctl_nlp_int_tap
out 224.0.0.0 240.0.0.0 ha_ctl_nlp_int_tap
out 255.255.255.255 255.255.255.255 ccl_ha_nlp_int_tap
out 224.0.0.0 240.0.0.0 ccl_ha_nlp_int_tap
out 255.255.255.255 255.255.255.255 nlp_int_tap
out 169.254.1.1 255.255.255.255 nlp_int_tap
out 169.254.1.0 255.255.255.248 nlp_int_tap
out 224.0.0.0 240.0.0.0 nlp_int_tap
out fd00:0:0:1::1 ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff nlp_int_tap
out fd00:0:0:1:: ffff:ffff:ffff:ffff:: nlp_int_tap
out fe80:: ffc0:: nlp_int_tap
out ff00:: ff00:: nlp_int_tap
out 0.0.0.0 0.0.0.0 via 0.0.0.0, identity
out :: :: via 0.0.0.0, identity

```

Key Points

The FTD (in a way similar to an Adaptive Security Appliance - ASA), first determines the exit (egress) interface of a packet (for that, it looks at the 'in' entries of the ASP routing table). Then for the determined interface, it tries to find the next-hop (for that, it looks at the 'out' entries of the ASP routing table). For example:

```

firepower# show asp table routing | include in.*198.51.100.0
in 198.51.100.0 255.255.255.248 via 192.0.2.99, OUTSIDE2
firepower#
firepower# show asp table routing | include out.*OUTSIDE2
out 255.255.255.255 255.255.255.255 OUTSIDE2
out 192.0.2.1 255.255.255.255 OUTSIDE2
out 198.51.100.0 255.255.255.248 via 192.0.2.99, OUTSIDE2
out 198.51.100.8 255.255.255.248 via 192.0.2.99, OUTSIDE2
out 198.51.100.16 255.255.255.248 via 192.0.2.99, OUTSIDE2
out 192.0.2.0 255.255.255.0 OUTSIDE2
out 224.0.0.0 240.0.0.0 OUTSIDE2

```

Finally, for the resolved next-hop the LINA checks the ARP cache for a valid adjacency.

The FTD packet-tracer tool confirms this process:

```

firepower# packet-tracer input INSIDE icmp 192.168.1.1 8 0 198.51.100.1

Phase: 1
Type: ACCESS-LIST
Subtype:
Result: ALLOW
Elapsed time: 7582 ns
Config:
Implicit Rule

```


Additional Information:

MAC Access list

Phase: 2

Type: INPUT-ROUTE-LOOKUP

Subtype: Resolve Egress Interface

Result: ALLOW

Elapsed time: 8474 ns

Config:

Additional Information:

Found next-hop 192.0.2.99 using egress ifc OUTSIDE2(vrfid:0)

Phase: 3

Type: ACCESS-LIST

Subtype: log

Result: ALLOW

Elapsed time: 5017 ns

Config:

access-group CSM_FW_ACL_ global

access-list CSM_FW_ACL_ advanced permit ip any any rule-id 268434433

access-list CSM_FW_ACL_ remark rule-id 268434433: ACCESS POLICY: mzafeiro_empty - Default

access-list CSM_FW_ACL_ remark rule-id 268434433: L4 RULE: DEFAULT ACTION RULE

Additional Information:

This packet will be sent to snort for additional processing where a verdict will be reached

Phase: 4

Type: CONN-SETTINGS

Subtype:

Result: ALLOW

Elapsed time: 5017 ns

Config:

class-map class-default

match any

policy-map global_policy

class class-default

set connection advanced-options UM_STATIC_TCP_MAP

service-policy global_policy global

Additional Information:

Phase: 5

Type: NAT

Subtype: per-session

Result: ALLOW

Elapsed time: 5017 ns

Config:

Additional Information:

Phase: 6

Type: IP-OPTIONS

Subtype:

Result: ALLOW

Elapsed time: 5017 ns

Config:

Additional Information:

Phase: 7

Type: INSPECT

Subtype: np-inspect

Result: ALLOW

Elapsed time: 57534 ns

Config:

class-map inspection_default

match default-inspection-traffic
policy-map global_policy
class inspection_default
inspect icmp
service-policy global_policy global
Additional Information:

Phase: 8
Type: INSPECT
Subtype: np-inspect
Result: ALLOW
Elapsed time: 3122 ns
Config:
Additional Information:

Phase: 9
Type: NAT
Subtype: per-session
Result: ALLOW
Elapsed time: 29882 ns
Config:
Additional Information:

Phase: 10
Type: IP-OPTIONS
Subtype:
Result: ALLOW
Elapsed time: 446 ns
Config:
Additional Information:

Phase: 11
Type: FLOW-CREATION
Subtype:
Result: ALLOW
Elapsed time: 20962 ns
Config:
Additional Information:
New flow created with id 178, packet dispatched to next module

Phase: 12
Type: EXTERNAL-INSPECT
Subtype:
Result: ALLOW
Elapsed time: 20070 ns
Config:
Additional Information:
Application: 'SNORT Inspect'

Phase: 13
Type: SNORT
Subtype:
Result: ALLOW
Elapsed time: 870592 ns
Config:
Additional Information:
Snort Trace:
Packet: ICMP
Session: new snort session
Snort id 1, NAP id 1, IPS id 0, Verdict PASS
Snort Verdict: (pass-packet) allow this packet

Phase: 14
Type: INPUT-ROUTE-LOOKUP-FROM-OUTPUT-ROUTE-LOOKUP
Subtype: Resolve Preferred Egress interface
Result: ALLOW
Elapsed time: 6244 ns
Config:
Additional Information:
Found next-hop 192.0.2.99 using egress ifc OUTSIDE2(vrfid:0)

Phase: 15
Type: ADJACENCY-LOOKUP
Subtype: Resolve Nexthop IP address to MAC
Result: ALLOW
Elapsed time: 1784 ns
Config:
Additional Information:
Found adjacency entry for Next-hop 192.0.2.99 on interface OUTSIDE2
Adjacency :Active
MAC address 4c4e.35fc.fcd8 hits 5 reference 1

Result:
input-interface: INSIDE(vrfid:0)
input-status: up
input-line-status: up
output-interface: OUTSIDE2(vrfid:0)
output-status: up
output-line-status: up
Action: allow
Time Taken: 1046760 ns

The FTD ARP table as it is seen in the Control Plane:

```
firepower# show arp
OUTSIDE1 203.0.113.99 4c4e.35fc.fcd8 3051
OUTSIDE2 192.0.2.99 4c4e.35fc.fcd8 5171
```

To force the ARP resolution:

```
firepower# ping 192.168.0.99
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.0.99, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
firepower# show arp
INSIDE 192.168.0.99 4c4e.35fc.fcd8 45
OUTSIDE1 203.0.113.99 4c4e.35fc.fcd8 32
OUTSIDE2 192.0.2.99 4c4e.35fc.fcd8 1
```

The FTD ARP table as it is seen in the Data Plane:

```

firepower# show asp table arp

Context: single_vf, Interface: OUTSIDE1
203.0.113.99 Active 4c4e.35fc.fcd8 hits 2 reference 1

Context: single_vf, Interface: OUTSIDE2
192.0.2.99 Active 4c4e.35fc.fcd8 hits 5 reference 0

Context: single_vf, Interface: INSIDE
192.168.0.99 Active 4c4e.35fc.fcd8 hits 5 reference 0

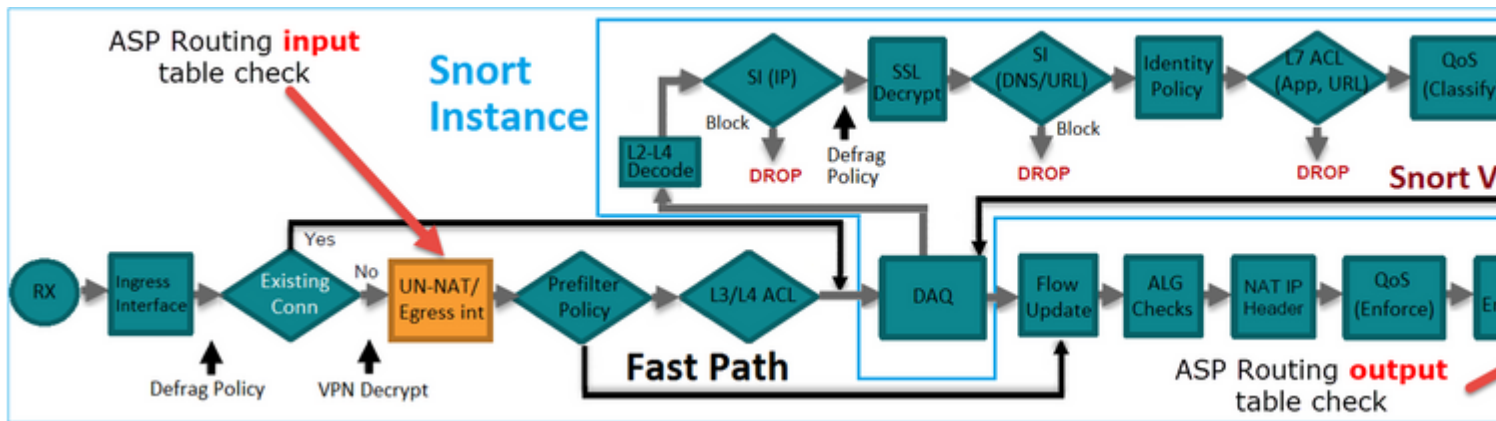
Context: single_vf, Interface: identity
:: Active 0000.0000.0000 hits 0 reference 0
0.0.0.0 Active 0000.0000.0000 hits 848 reference 0

Last clearing of hits counters: Never

```

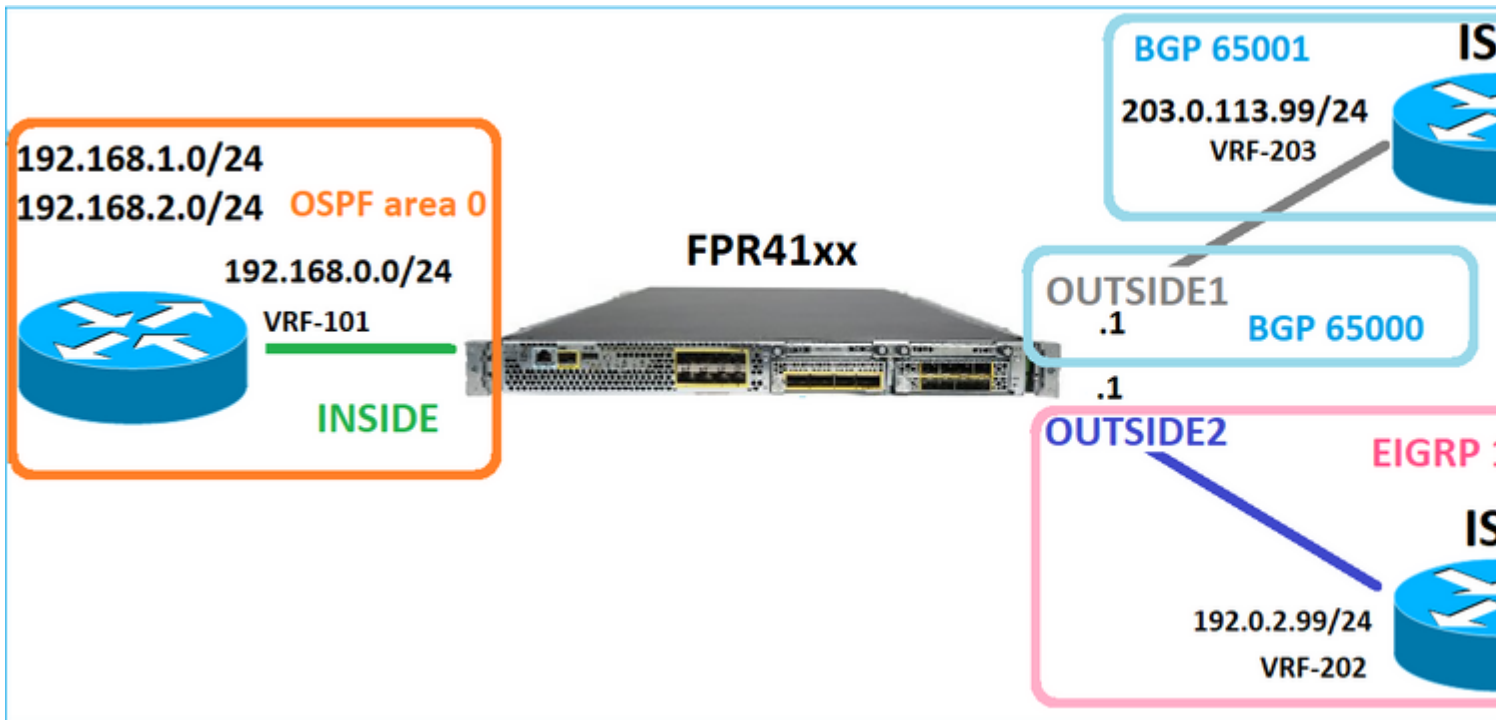
FTD Order of Operations

The image shows the order of operations and where the input and output ASP Routing checks are done:



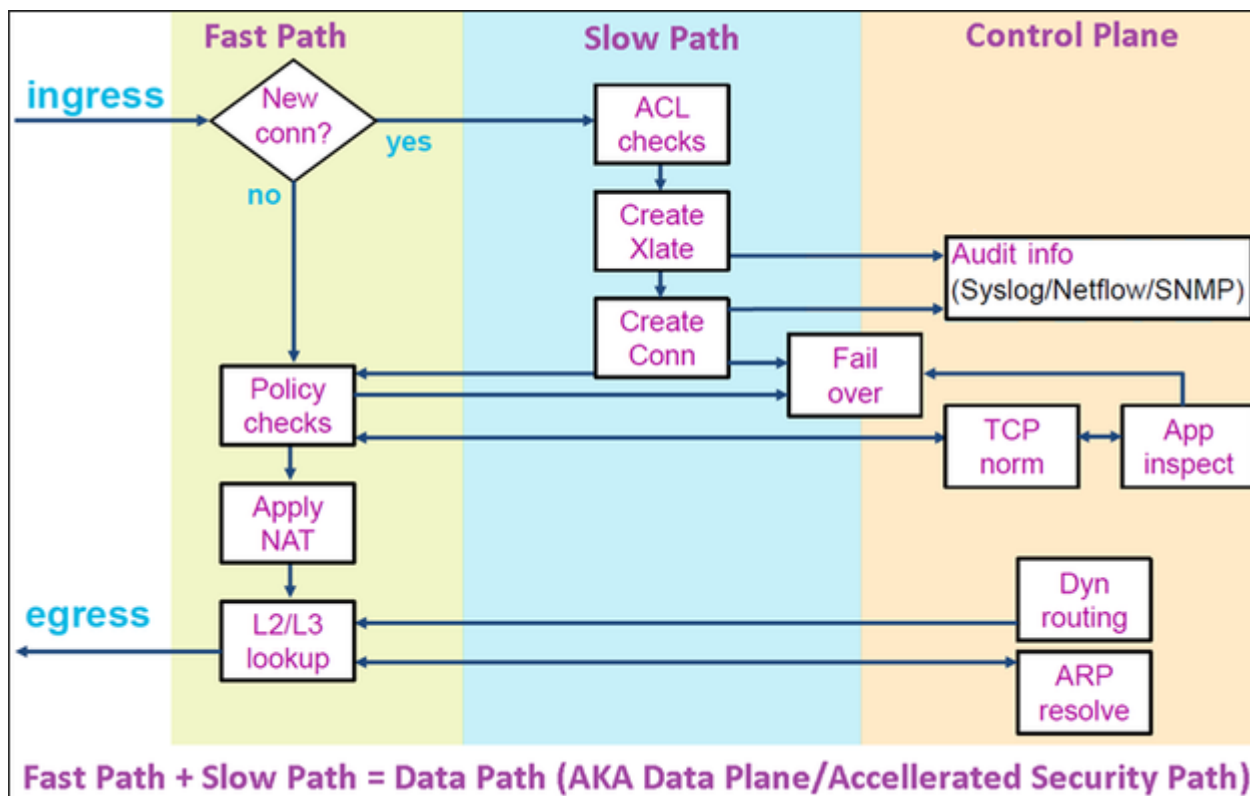
Configure

Case 1 – Forwarding Based on Connection Lookup



As was mentioned already, the main component of the FTD LINA Engine is the Datapath process (multiple instances based on the number of device cores). Furthermore, the Datapath (also known as Accelerated Security Path – ASP) consists of 2 Paths:

1. Slow Path = Responsible for new connection establishment (it populates the Fast Path).
2. Fast Path = Handles packets that belong to established connections.



- Commands like show route and show arp show the contents of the Control Plane.
- On the other hand, commands like show asp table routing and show asp table arp show the contents of ASP (Datapath) which is what is actually applied.

Enable capture with trace on FTD INSIDE interface:

```
firepower# capture CAPI trace detail interface INSIDE match ip host 192.168.1.1 host 198.51.100.1
```

Open a Telnet session through the FTD:

```
Router1# telnet 198.51.100.1 /vrf VRF-101 /source-interface lo1
Trying 198.51.100.1 ... Open
```

The FTD captures show the packets from the beginning of the connection (TCP 3-way handshake is captured):

```
firepower# show capture CAPI
```

26 packets captured

```
1: 10:50:38.407190 802.1Q vlan#101 P0 192.168.1.1.57734 > 198.51.100.1.23: S 1306692135:1306692135(0) wi
2: 10:50:38.408929 802.1Q vlan#101 P0 198.51.100.1.23 > 192.168.1.1.57734: S 1412677784:1412677784(0) ac
3: 10:50:38.409265 802.1Q vlan#101 P0 192.168.1.1.57734 > 198.51.100.1.23: . ack 1412677785 win 4128
4: 10:50:38.409433 802.1Q vlan#101 P0 192.168.1.1.57734 > 198.51.100.1.23: P 1306692136:1306692154(18) a
5: 10:50:38.409845 802.1Q vlan#101 P0 192.168.1.1.57734 > 198.51.100.1.23: . ack 1412677785 win 4128
6: 10:50:38.410135 802.1Q vlan#101 P0 198.51.100.1.23 > 192.168.1.1.57734: . ack 1306692154 win 4110
7: 10:50:38.411355 802.1Q vlan#101 P0 198.51.100.1.23 > 192.168.1.1.57734: P 1412677785:1412677797(12) a
8: 10:50:38.413049 802.1Q vlan#101 P0 192.168.1.1.57734 > 198.51.100.1.23: P 1306692154:1306692157(3) ac
9: 10:50:38.413140 802.1Q vlan#101 P0 192.168.1.1.57734 > 198.51.100.1.23: P 1306692157:1306692166(9) ac
10: 10:50:38.414071 802.1Q vlan#101 P0 198.51.100.1.23 > 192.168.1.1.57734: . 1412677797:1412678322(525)
...
```

Trace the first packet (TCP SYN). This packet goes through the FTD LINA Slow Path, and a Global Routing lookup is done in this case:

```
firepower# show capture CAPI packet-number 1 trace
```

26 packets captured

```
1: 10:50:38.407190 802.1Q vlan#101 P0 192.168.1.1.57734 > 198.51.100.1.23: S 1306692135:1306692135(0)
Phase: 1
Type: CAPTURE
Subtype:
Result: ALLOW
Elapsed time: 4683 ns
Config:
Additional Information:
Forward Flow based lookup yields rule:
in id=0x1505f1d17940, priority=13, domain=capture, deny=false
```

hits=1783, user_data=0x1505f2096910, cs_id=0x0, l3_type=0x0
src mac=0000.0000.0000, mask=0000.0000.0000
dst mac=0000.0000.0000, mask=0000.0000.0000
input_ifc=INSIDE, output_ifc=any

Phase: 2

Type: ACCESS-LIST

Subtype:

Result: ALLOW

Elapsed time: 4683 ns

Config:

Implicit Rule

Additional Information:

Forward Flow based lookup yields rule:

in id=0x1502a7ba4d40, priority=1, domain=permit, deny=false

hits=28, user_data=0x0, cs_id=0x0, l3_type=0x8

src mac=0000.0000.0000, mask=0000.0000.0000

dst mac=0000.0000.0000, mask=0100.0000.0000

input_ifc=INSIDE, output_ifc=any

Phase: 3

Type: INPUT-ROUTE-LOOKUP

Subtype: Resolve Egress Interface

Result: ALLOW

Elapsed time: 5798 ns

Config:

Additional Information:

Found next-hop 192.0.2.99 using egress ifc OUTSIDE2(vrfid:0)

Phase: 4

Type: ACCESS-LIST

Subtype: log

Result: ALLOW

Elapsed time: 3010 ns

Config:

access-group CSM_FW_ACL_ global

access-list CSM_FW_ACL_ advanced permit ip any any rule-id 268434433

access-list CSM_FW_ACL_ remark rule-id 268434433: ACCESS POLICY: mzafeiro_empty - Default

access-list CSM_FW_ACL_ remark rule-id 268434433: L4 RULE: DEFAULT ACTION RULE

Additional Information:

This packet will be sent to snort for additional processing where a verdict will be reached

Forward Flow based lookup yields rule:

in id=0x1505f1e2e980, priority=12, domain=permit, deny=false

hits=4, user_data=0x15024a56b940, cs_id=0x0, use_real_addr, flags=0x0, protocol=0

src ip/id=0.0.0.0, mask=0.0.0.0, port=0, tag=any, ifc=any

dst ip/id=0.0.0.0, mask=0.0.0.0, port=0, tag=any, ifc=any,, dscp=0x0, nsg_id=none

input_ifc=any, output_ifc=any

Phase: 5

Type: CONN-SETTINGS

Subtype:

Result: ALLOW

Elapsed time: 3010 ns

Config:

class-map class-default

match any

policy-map global_policy

class class-default

set connection advanced-options UM_STATIC_TCP_MAP

service-policy global_policy global

Additional Information:

Forward Flow based lookup yields rule:

in id=0x1505f1f18bc0, priority=7, domain=conn-set, deny=false
hits=4, user_data=0x1505f1f13f70, cs_id=0x0, use_real_addr, flags=0x0, protocol=0
src ip/id=0.0.0.0, mask=0.0.0.0, port=0, tag=any
dst ip/id=0.0.0.0, mask=0.0.0.0, port=0, tag=any, dscp=0x0, nsg_id=none
input_ifc=INSIDE(vrfid:0), output_ifc=any

Phase: 6

Type: NAT

Subtype: per-session

Result: ALLOW

Elapsed time: 3010 ns

Config:

Additional Information:

Forward Flow based lookup yields rule:

in id=0x15052e96b150, priority=0, domain=nat-per-session, deny=false
hits=125, user_data=0x0, cs_id=0x0, reverse, use_real_addr, flags=0x0, protocol=6
src ip/id=0.0.0.0, mask=0.0.0.0, port=0, tag=any
dst ip/id=0.0.0.0, mask=0.0.0.0, port=0, tag=any, dscp=0x0, nsg_id=none
input_ifc=any, output_ifc=any

Phase: 7

Type: IP-OPTIONS

Subtype:

Result: ALLOW

Elapsed time: 3010 ns

Config:

Additional Information:

Forward Flow based lookup yields rule:

in id=0x1502a7bacde0, priority=0, domain=inspect-ip-options, deny=true
hits=19, user_data=0x0, cs_id=0x0, reverse, flags=0x0, protocol=0
src ip/id=0.0.0.0, mask=0.0.0.0, port=0, tag=any
dst ip/id=0.0.0.0, mask=0.0.0.0, port=0, tag=any, dscp=0x0, nsg_id=none
input_ifc=INSIDE(vrfid:0), output_ifc=any

Phase: 8

Type: NAT

Subtype: per-session

Result: ALLOW

Elapsed time: 52182 ns

Config:

Additional Information:

Reverse Flow based lookup yields rule:

in id=0x15052e96b150, priority=0, domain=nat-per-session, deny=false
hits=127, user_data=0x0, cs_id=0x0, reverse, use_real_addr, flags=0x0, protocol=6
src ip/id=0.0.0.0, mask=0.0.0.0, port=0, tag=any
dst ip/id=0.0.0.0, mask=0.0.0.0, port=0, tag=any, dscp=0x0, nsg_id=none
input_ifc=any, output_ifc=any

Phase: 9

Type: IP-OPTIONS

Subtype:

Result: ALLOW

Elapsed time: 892 ns

Config:

Additional Information:

Reverse Flow based lookup yields rule:

in id=0x1502a7f9b460, priority=0, domain=inspect-ip-options, deny=true
hits=38, user_data=0x0, cs_id=0x0, reverse, flags=0x0, protocol=0
src ip/id=0.0.0.0, mask=0.0.0.0, port=0, tag=any
dst ip/id=0.0.0.0, mask=0.0.0.0, port=0, tag=any, dscp=0x0, nsg_id=none
input_ifc=OUTSIDE2(vrfid:0), output_ifc=any

Phase: 10
Type: FLOW-CREATION
Subtype:
Result: ALLOW
Elapsed time: 25422 ns
Config:
Additional Information:
New flow created with id 244, packet dispatched to next module
Module information for forward flow ...
snp_fp_inspect_ip_options
snp_fp_tcp_normalizer
snp_fp_tcp_proxy
snp_fp_snort
snp_fp_tcp_proxy
snp_fp_translate
snp_fp_tcp_normalizer
snp_fp_adjacency
snp_fp_fragment
snp_ifc_stat

Module information for reverse flow ...
snp_fp_inspect_ip_options
snp_fp_tcp_normalizer
snp_fp_translate
snp_fp_tcp_proxy
snp_fp_snort
snp_fp_tcp_proxy
snp_fp_tcp_normalizer
snp_fp_adjacency
snp_fp_fragment
snp_ifc_stat

Phase: 11
Type: EXTERNAL-INSPECT
Subtype:
Result: ALLOW
Elapsed time: 36126 ns
Config:
Additional Information:
Application: 'SNORT Inspect'

Phase: 12
Type: SNORT
Subtype:
Result: ALLOW
Elapsed time: 564636 ns
Config:
Additional Information:
Snort Trace:
Packet: TCP, SYN, seq 182318660
Session: new snort session
AppID: service unknown (0), application unknown (0)
Snort id 28, NAP id 1, IPS id 0, Verdict PASS
Snort Verdict: (pass-packet) allow this packet

Phase: 13
Type: INPUT-ROUTE-LOOKUP-FROM-OUTPUT-ROUTE-LOOKUP
Subtype: Resolve Preferred Egress interface
Result: ALLOW
Elapsed time: 7136 ns
Config:
Additional Information:

Found next-hop 192.0.2.99 using egress ifc OUTSIDE2(vrfid:0)

Phase: 14

Type: ADJACENCY-LOOKUP

Subtype: Resolve Nexthop IP address to MAC

Result: ALLOW

Elapsed time: 2230 ns

Config:

Additional Information:

Found adjacency entry for Next-hop 192.0.2.99 on interface OUTSIDE2

Adjacency :Active

MAC address 4c4e.35fc.fcd8 hits 10 reference 1

Phase: 15

Type: CAPTURE

Subtype:

Result: ALLOW

Elapsed time: 5352 ns

Config:

Additional Information:

Forward Flow based lookup yields rule:

out id=0x150521389870, priority=13, domain=capture, deny=false

hits=1788, user_data=0x1505f1d2b630, cs_id=0x0, l3_type=0x0

src mac=0000.0000.0000, mask=0000.0000.0000

dst mac=0000.0000.0000, mask=0000.0000.0000

input_ifc=OUTSIDE2, output_ifc=any

Result:

input-interface: INSIDE(vrfid:0)

input-status: up

input-line-status: up

output-interface: OUTSIDE2(vrfid:0)

output-status: up

output-line-status: up

Action: allow

Time Taken: 721180 ns

1 packet shown

firepower#

Trace another ingress packet from the same flow. The packet that matches an active connection:

firepower# show capture CAPI packet-number 3 trace

33 packets captured

3: 10:50:38.409265 802.1Q vlan#101 P0 192.168.1.1.57734 > 198.51.100.1.23: . ack 1412677785 win 4128

Phase: 1

Type: CAPTURE

Subtype:

Result: ALLOW

Elapsed time: 2676 ns

Config:

Additional Information:

Forward Flow based lookup yields rule:

in id=0x1505f1d17940, priority=13, domain=capture, deny=false

hits=105083, user_data=0x1505f2096910, cs_id=0x0, l3_type=0x0

src mac=0000.0000.0000, mask=0000.0000.0000
dst mac=0000.0000.0000, mask=0000.0000.0000
input_ifc=INSIDE, output_ifc=any

Phase: 2

Type: ACCESS-LIST

Subtype:

Result: ALLOW

Elapsed time: 2676 ns

Config:

Implicit Rule

Additional Information:

Forward Flow based lookup yields rule:

in id=0x1502a7ba4d40, priority=1, domain=permit, deny=false

hits=45, user_data=0x0, cs_id=0x0, l3_type=0x8

src mac=0000.0000.0000, mask=0000.0000.0000

dst mac=0000.0000.0000, mask=0100.0000.0000

input_ifc=INSIDE, output_ifc=any

Phase: 3

Type: FLOW-LOOKUP

Subtype:

Result: ALLOW

Elapsed time: 1338 ns

Config:

Additional Information:

Found flow with id 2552, using existing flow

Module information for forward flow ...

snp_fp_inspect_ip_options

snp_fp_tcp_normalizer

snp_fp_snort

snp_fp_translate

snp_fp_tcp_normalizer

snp_fp_adjacency

snp_fp_fragment

snp_ifc_stat

Module information for reverse flow ...

snp_fp_inspect_ip_options

snp_fp_tcp_normalizer

snp_fp_translate

snp_fp_snort

snp_fp_tcp_normalizer

snp_fp_adjacency

snp_fp_fragment

snp_ifc_stat

Phase: 4

Type: EXTERNAL-INSPECT

Subtype:

Result: ALLOW

Elapsed time: 16502 ns

Config:

Additional Information:

Application: 'SNORT Inspect'

Phase: 5

Type: SNORT

Subtype:

Result: ALLOW

Elapsed time: 12934 ns

Config:

Additional Information:

Snort Trace:

Packet: TCP, ACK, seq 1306692136, ack 1412677785
AppID: service unknown (0), application unknown (0)
Snort id 19, NAP id 1, IPS id 0, Verdict PASS
Snort Verdict: (pass-packet) allow this packet

Result:

input-interface: INSIDE(vrfid:0)
input-status: up
input-line-status: up
Action: allow
Time Taken: 36126 ns

1 packet shown
firepower#

Floating Timeout

The Problem

Temporary route instability can cause long-lived (elephant) UDP connections through the FTD to be established through different FTD interfaces than desired.

The Solution

To remediate this, set the timeout floating-conn to a value different than the default which is disabled:



FTD4100-1

Enter Description

- ARP Inspection
- Banner
- DNS
- External Authentication
- Fragment Settings
- HTTP Access
- ICMP Access
- SSH Access
- SMTP Server
- SNMP
- SSL
- Syslog
- Timeouts**
- Time Synchronization
- Time Zone
- UCAPL/CC Compliance

Console Timeout*	<input type="text" value="0"/>	(0 - 1440 mins)	?
Translation Slot(xlate)	<input type="text" value="Default"/>	3:00:00	(3:0:0 or 0:1:0 - 1193:0:0)
Connection(Conn)	<input type="text" value="Default"/>	1:00:00	(0:0:0 or 0:5:0 - 1193:0:0)
Half-Closed	<input type="text" value="Default"/>	0:10:00	(0:0:0 or 0:0:30 - 1193:0:0)
UDP	<input type="text" value="Default"/>	0:02:00	(0:0:0 or 0:1:0 - 1193:0:0)
ICMP	<input type="text" value="Default"/>	0:00:02	(0:0:2 or 0:0:2 - 1193:0:0)
RPC/Sun RPC	<input type="text" value="Default"/>	0:10:00	(0:0:0 or 0:1:0 - 1193:0:0)
H.225	<input type="text" value="Default"/>	1:00:00	(0:0:0 or 0:0:0 - 1193:0:0)
H.323	<input type="text" value="Default"/>	0:05:00	(0:0:0 or 0:0:0 - 1193:0:0)
SIP	<input type="text" value="Default"/>	0:30:00	(0:0:0 or 0:5:0 - 1193:0:0)
SIP Media	<input type="text" value="Default"/>	0:02:00	(0:0:0 or 0:1:0 - 1193:0:0)
SIP Disconnect:	<input type="text" value="Default"/>	0:02:00	(0:02:0 or 0:0:1 - 0:10:0)
SIP Invite	<input type="text" value="Default"/>	0:03:00	(0:1:0 or 0:1:0 - 0:30:0)
SIP Provisional Media	<input type="text" value="Default"/>	0:02:00	(0:2:0 or 0:1:0 - 0:30:0)
Floating Connection	<input type="text" value="Default"/>	0:00:00	(0:0:0 or 0:0:30 - 1193:0:0)
Xlate-PAT	<input type="text" value="Default"/>	0:00:30	(0:0:30 or 0:0:30 - 0:5:0)

From the Command Reference:

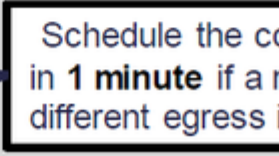
floating-conn When multiple routes exist to a network with different metrics, the ASA uses the one with the best metric at the time of connection creation. If a better route becomes available, then this timeout lets connections be closed so a connection can be reestablished to use the better route. The default is 0 (the connection never times out). To make it possible to use better routes, set the timeout to a value between 0:0:30 and 1193:0:0.

For more details, see Case Study: UDP Connections Fail After Reload from the CiscoLive BRKSEC-3020 session:

Floating Connection Timeout

- The “bad” connection never times out since the UDP traffic
 - TCP is stateful, so the connection would terminate and re-establish
 - ASA needs to tear the original connection down when the connection is replaced
 - ASA 8.4(2)+ introduces **timeout floating-conn** to accomplish this

```
asa# show run timeout
timeout xlate 9:00:00
timeout pat-xlate 0:00:30
timeout conn 1:00:00 half-closed 0:10:00 udp 0:02:00 icmp 0:00:02
timeout sunrpc 0:10:00 h323 0:05:00 h225 1:00:00 mgcp 0:05:00 mgcp-
timeout sip 0:30:00 sip_media 0:02:00 sip-invite 0:03:00 sip-discover
timeout sip-provisional-media 0:02:00 uauth 9:00:00 absolute uauth
timeout tcp-proxy-reassembly 0:01:00
timeout floating-conn 0:00:00
asa#
asa# configure terminal
asa(config)# timeout floating-conn 0:01:00
```



Schedule the connection to timeout in **1 minute** if a new connection is established on a different egress interface.

Conn-holddown Timeout

The Problem

A route goes down (is removed), but the traffic matches an established connection.

The Solution

Timeout conn-holddown feature was added on ASA 9.6.2. The feature is enabled by default, but currently (7.1.x) is unsupported by FMC UI or FlexConfig. Related enhancement: [ENH: timeout conn-holddown not available for configuration in FMC](#)

From the ASA CLI guide:

conn-holddown	How long the system should maintain a connection when the route used by the connection no longer exists or is inactive. If the route does not become active within this holddown period, the connection is freed. The purpose of the connection holddown timer is to reduce the effect of route flapping, where routes might come up and go down quickly. You can reduce the holddown timer to make route convergence happen more quickly. The default is 15 seconds, the range is 00:00:00 to 00:00:15.
----------------------	--

```
firepower# show run all timeout
timeout xlate 3:00:00
timeout pat-xlate 0:00:30
timeout conn 1:00:00 half-closed 0:10:00 udp 0:02:00 sctp 0:02:00 icmp 0:00:02
```



```

timeout sunrpc 0:10:00 h323 0:05:00 h225 1:00:00 mgcp 0:05:00 mgcp-pat 0:05:00
timeout sip 0:30:00 sip_media 0:02:00 sip-invite 0:03:00 sip-disconnect 0:02:00
timeout sip-provisional-media 0:02:00 uauth 0:05:00 absolute
timeout tcp-proxy-reassembly 0:00:30
timeout floating-conn 0:00:00
timeout conn-holddown 0:00:15
timeout igp stale-route 0:01:10

```

Case 2 – Forwarding Based on NAT Lookup

Requirement

Configure this NAT rule:

- Type: Static
- Source Interface: INSIDE
- Destination Interface: OUTSIDE1
- Original Source: 192.168.1.1
- Original Destination: 198.51.100.1
- Translated Source: 192.168.1.1
- Translated Destination: 198.51.100.1

Solution

						Original Packet				
<input type="checkbox"/>	#	Direction	Type	Source Interface Objects	Destination Interface Objects	Original Sources	Original Destinations	Original Services	Translated Sources	Translated Destinations
NAT Rules Before										
<input type="checkbox"/>	1	#	Static	INSIDE_FTD4100-1	OUTSIDE1_FTD4100	host_192.168.1.1	host_198.51.100.1		host_192.168.1.1	host_198.51.100.1
Auto NAT Rules										

The deployed NAT rule on the FTD CLI:

```

firepower# show run nat
nat (INSIDE,OUTSIDE1) source static host_192.168.1.1 host_192.168.1.1 destination static host_198.51.100.1 host_198.51.100.1
firepower# show nat
Manual NAT Policies (Section 1)
1 (INSIDE) to (OUTSIDE1) source static host_192.168.1.1 host_192.168.1.1 destination static host_198.51.100.1 host_198.51.100.1
translate_hits = 0, untranslate_hits = 0

```

Configure 3 captures:

```
firepower# capture CAPI trace detail interface INSIDE match ip host 192.168.1.1 host 198.51.100.1
firepower# capture CAP01 interface OUTSIDE1 match ip host 192.168.1.1 any
firepower# capture CAP02 interface OUTSIDE2 match ip host 192.168.1.1 any
firepower# show capture
capture CAPI type raw-data trace detail interface INSIDE [Capturing - 0 bytes]
match ip host 192.168.1.1 host 198.51.100.1
capture CAP01 type raw-data interface OUTSIDE1 [Capturing - 0 bytes]
match ip host 192.168.1.1 any
capture CAP02 type raw-data interface OUTSIDE2 [Capturing - 0 bytes]
match ip host 192.168.1.1 any
```

Initiate a telnet session from 192.168.1.1 to 198.51.100.1:

```
Router1# telnet 198.51.100.1 /vrf VRF-101 /source-interface lo1
Trying 198.51.100.1 ...
% Connection timed out; remote host not responding
```

Packets arrive on FTD, but nothing leaves OUTSIDE1 nor OUTSIDE2 interfaces:

```
firepower# show capture
capture CAPI type raw-data trace detail interface INSIDE [Capturing - 156 bytes]
match ip host 192.168.1.1 host 198.51.100.1
capture CAP01 type raw-data interface OUTSIDE1 [Capturing - 0 bytes]
match ip host 192.168.1.1 any
capture CAP02 type raw-data interface OUTSIDE2 [Capturing - 0 bytes]
match ip host 192.168.1.1 any
```

Trace the TCP SYN packet. Phase 3 (UN-NAT) shows that NAT (UN-NAT specifically) diverted the packet to the OUTSIDE1 interface for next-hop lookup:

```
firepower# show capture CAPI
2 packets captured
1: 11:22:59.179678 802.1Q vlan#101 P0 192.168.1.1.38790 > 198.51.100.1.23: S 1174675193:1174675193(0) wi
2: 11:23:01.179632 802.1Q vlan#101 P0 192.168.1.1.38790 > 198.51.100.1.23: S 1174675193:1174675193(0) wi
2 packets shown
firepower#
```

```
firepower# show capture CAPI packet-number 1 trace detail
```

```
2 packets captured
1: 11:22:59.179678 4c4e.35fc.fcd8 00be.75f6.1dae 0x8100 Length: 62
802.1Q vlan#101 P0 192.168.1.1.38790 > 198.51.100.1.23: S [tcp sum ok] 1174675193:1174675193(0) win 4128
...
```

Phase: 3
Type: UN-NAT
Subtype: static
Result: ALLOW
Elapsed time: 6244 ns
Config:
nat (INSIDE,OUTSIDE1) source static host_192.168.1.1 host_192.168.1.1 destination static host_198.51.100.1
Additional Information:
NAT divert to egress interface OUTSIDE1(vrfid:0)
Untranslate 198.51.100.1/23 to 198.51.100.1/23

...
Phase: 12
Type: FLOW-CREATION
Subtype:
Result: ALLOW
Elapsed time: 25422 ns
Config:
Additional Information:
New flow created with id 2614, packet dispatched to next module
Module information for forward flow ...
snf_fp_inspect_ip_options
snf_fp_tcp_normalizer
snf_fp_tcp_proxy
snf_fp_snort
snf_fp_tcp_proxy
snf_fp_translate
snf_fp_tcp_normalizer
snf_fp_adjacency
snf_fp_fragment
snf_ifc_stat

Phase: 15
Type: INPUT-ROUTE-LOOKUP-FROM-OUTPUT-ROUTE-LOOKUP
Subtype: Resolve Preferred Egress interface
Result: ALLOW
Elapsed time: 8028 ns
Config:
Additional Information:
Found next-hop 192.0.2.99 using egress ifc OUTSIDE2(vrfid:0)

Phase: 16
Type: SUBOPTIMAL-LOOKUP
Subtype: suboptimal next-hop
Result: ALLOW
Elapsed time: 446 ns
Config:
Additional Information:
Input route lookup returned ifc OUTSIDE2 is not same as existing ifc OUTSIDE1

Result:
input-interface: INSIDE(vrfid:0)
input-status: up
input-line-status: up
output-interface: OUTSIDE1(vrfid:0)
output-status: up
output-line-status: up
Action: drop
Time Taken: 777375 ns
Drop-reason: (no-adjacency) No valid adjacency, Drop-location: frame 0x00005577204a7287 flow (NA)/NA

1 packet shown

In this case, the SUBOPTIMAL-LOOKUP means that the egress interface determined by the NAT process (OUTSIDE1) is different than the egress interface specified in the ASP input table:

```
firepower# show asp table routing | include 198.51.100.0
in 198.51.100.0 255.255.255.248 via 192.0.2.99, OUTSIDE2
out 198.51.100.0 255.255.255.248 via 192.0.2.99, OUTSIDE2
```

A possible workaround is to add a floating static route on the OUTSIDE1 interface:

```
firepower# show run route
route OUTSIDE2 198.51.100.0 255.255.255.248 192.0.2.99 1
route OUTSIDE1 198.51.100.0 255.255.255.248 203.0.113.99 200
```

Note: If you try to add a static route with the same metric as the one that already exists, this error appears:

The screenshot shows the Cisco Firepower GUI with the 'Routing' tab selected. The left sidebar shows 'Manage Virtual Routers' with 'Global' selected. The main area displays a routing table with the following entries:

Network	Interface	Leaked from Virtual Router
IPv4 Routes		
net_198.51.100.0_29bits	OUTSIDE1	
net_198.51.100.0_29bits	OUTSIDE2	
IPv6 Routes		

An error message is displayed in a white box on the right:

```
Error - Device Configuration
Virtual router [Global] - Invalid IPv4
The interfaces OUTSIDE2,OUTSIDE1
network address 198.51.100.0/29 are
considered as ECMP eligible routes.
Please Configure ECMP with above i
```

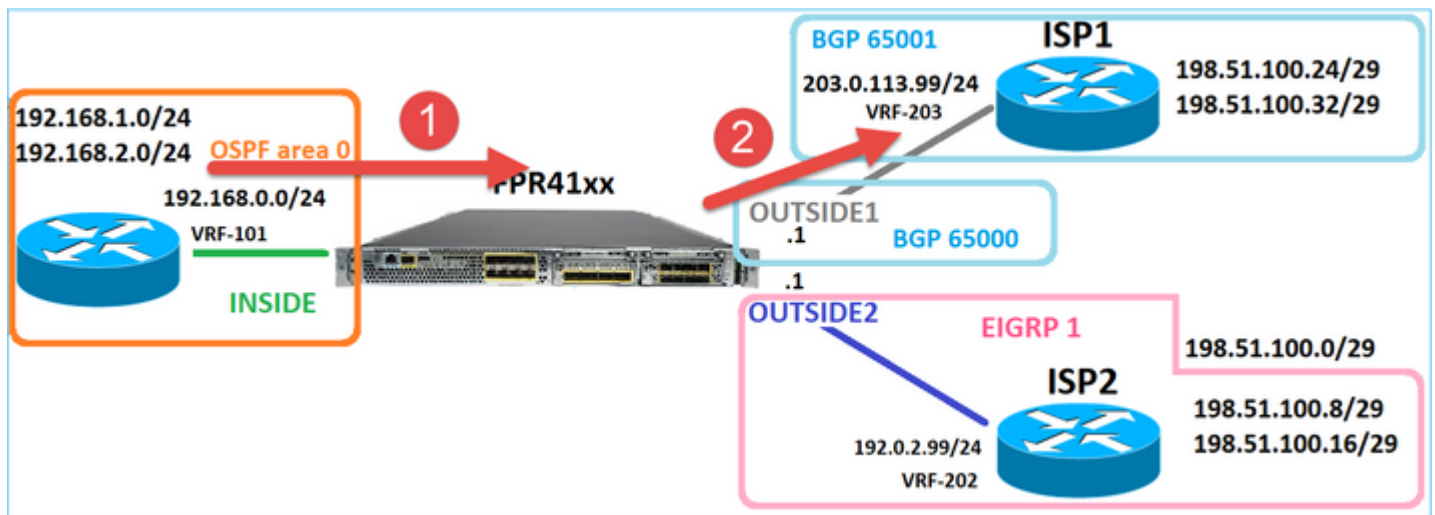
Note: Floating route with a Distance Metric of 255 is not installed in the routing table.

Try to Telnet that there are packets sent through the FTD:

```
Router1# telnet 198.51.100.1 /vrf VRF-101 /source-interface lo1
Trying 198.51.100.1 ...
% Connection timed out; remote host not responding
```

```
firepower# show capture
capture CAPI type raw-data trace detail interface INSIDE [Capturing - 156 bytes]
match ip host 192.168.1.1 host 198.51.100.1
capture CAP01 type raw-data interface OUTSIDE1 [Capturing - 312 bytes]
match ip host 192.168.1.1 any
capture CAP02 type raw-data interface OUTSIDE2 [Capturing - 386 bytes]
match ip host 192.168.1.1 any
```

Packet trace shows that the packets are forwarded to ISP1 (OUTSIDE1) interface instead of ISP2 due to NAT Lookup:



```
firepower# show capture CAPI packet-number 1 trace
```

```
2 packets captured
```

```
1: 09:03:02.773962 802.1Q vlan#101 P0 192.168.1.1.16774 > 198.51.100.1.23: S 2910053251:2910053251(0) w
...
```

```
Phase: 3
```

```
Type: UN-NAT
```

```
Subtype: static
```

```
Result: ALLOW
```

```
Elapsed time: 4460 ns
```

```
Config:
```

```
nat (INSIDE,OUTSIDE1) source static host_192.168.1.1 host_192.168.1.1 destination static host_198.51.100.1
```

```
Additional Information:
```

```
NAT divert to egress interface OUTSIDE1(vrfid:0)
```

```
Untranslate 198.51.100.1/23 to 198.51.100.1/23
```

```
...
```

```
Phase: 12
```

Type: FLOW-CREATION
Subtype:
Result: ALLOW
Elapsed time: 29436 ns
Config:
Additional Information:
New flow created with id 2658, packet dispatched to next module
Module information for forward flow ...
snf_fp_inspect_ip_options
snf_fp_tcp_normalizer
snf_fp_snort
snf_fp_translate
snf_fp_tcp_normalizer
snf_fp_adjacency
snf_fp_fragment
snf_ifc_stat

Phase: 15
Type: INPUT-ROUTE-LOOKUP-FROM-OUTPUT-ROUTE-LOOKUP
Subtype: Resolve Preferred Egress interface
Result: ALLOW
Elapsed time: 5798 ns
Config:
Additional Information:
Found next-hop 192.0.2.99 using egress ifc OUTSIDE2(vrfid:0)

Phase: 16
Type: SUBOPTIMAL-LOOKUP
Subtype: suboptimal next-hop
Result: ALLOW
Elapsed time: 446 ns
Config:
Additional Information:
Input route lookup returned ifc OUTSIDE2 is not same as existing ifc OUTSIDE1

Phase: 17
Type: NEXTHOP-LOOKUP-FROM-OUTPUT-ROUTE-LOOKUP
Subtype: Lookup Nexthop on interface
Result: ALLOW
Elapsed time: 1784 ns
Config:
Additional Information:
Found next-hop 203.0.113.99 using egress ifc OUTSIDE1(vrfid:0)

Phase: 18
Type: ADJACENCY-LOOKUP
Subtype: Resolve Nexthop IP address to MAC
Result: ALLOW
Elapsed time: 1338 ns
Config:
Additional Information:
Found adjacency entry for Next-hop 203.0.113.99 on interface OUTSIDE1
Adjacency :Active
MAC address 4c4e.35fc.fcd8 hits 106 reference 2
...

Result:
input-interface: INSIDE(vrfid:0)
input-status: up
input-line-status: up
output-interface: OUTSIDE1(vrfid:0)
output-status: up

```
output-line-status: up
Action: allow
Time Taken: 723409 ns
```

```
1 packet shown
firepower#
```

Interestingly, in this case, there are packets shown on INSIDE and both egress interfaces:

```
firepower# show capture CAPI
```

```
2 packets captured
```

```
1: 09:03:02.773962 802.1Q vlan#101 P0 192.168.1.1.32134 > 198.51.100.1.23: S 3031010184:3031010184(0) wi
2: 09:03:05.176565 802.1Q vlan#101 P0 192.168.1.1.32134 > 198.51.100.1.23: S 3031010184:3031010184(0) wi
```

```
2 packets shown
```

```
firepower# show capture CAP01
```

```
4 packets captured
```

```
1: 09:03:02.774358 802.1Q vlan#203 P0 192.168.1.1.32134 > 198.51.100.1.23: S 3249840142:3249840142(0) wi
2: 09:03:02.774557 802.1Q vlan#203 P0 192.168.1.1.32134 > 198.51.100.1.23: S 3249840142:3249840142(0) wi
3: 09:03:05.176702 802.1Q vlan#203 P0 192.168.1.1.32134 > 198.51.100.1.23: S 3249840142:3249840142(0) wi
4: 09:03:05.176870 802.1Q vlan#203 P0 192.168.1.1.32134 > 198.51.100.1.23: S 3249840142:3249840142(0) wi
```

```
4 packets shown
```

```
firepower# show capture CAP02
```

```
5 packets captured
```

```
1: 09:03:02.774679 802.1Q vlan#202 P0 192.168.1.1.32134 > 198.51.100.1.23: S 194652172:194652172(0) win
2: 09:03:02.775457 802.1Q vlan#202 P0 198.51.100.1.23 > 192.168.1.1.32134: S 4075003210:4075003210(0) ac
3: 09:03:05.176931 802.1Q vlan#202 P0 192.168.1.1.32134 > 198.51.100.1.23: S 194652172:194652172(0) win
4: 09:03:05.177282 802.1Q vlan#202 P0 198.51.100.1.23 > 192.168.1.1.32134: . ack 194652173 win 4128
5: 09:03:05.180517 802.1Q vlan#202 P0 198.51.100.1.23 > 192.168.1.1.32134: S 4075003210:4075003210(0) ac
```

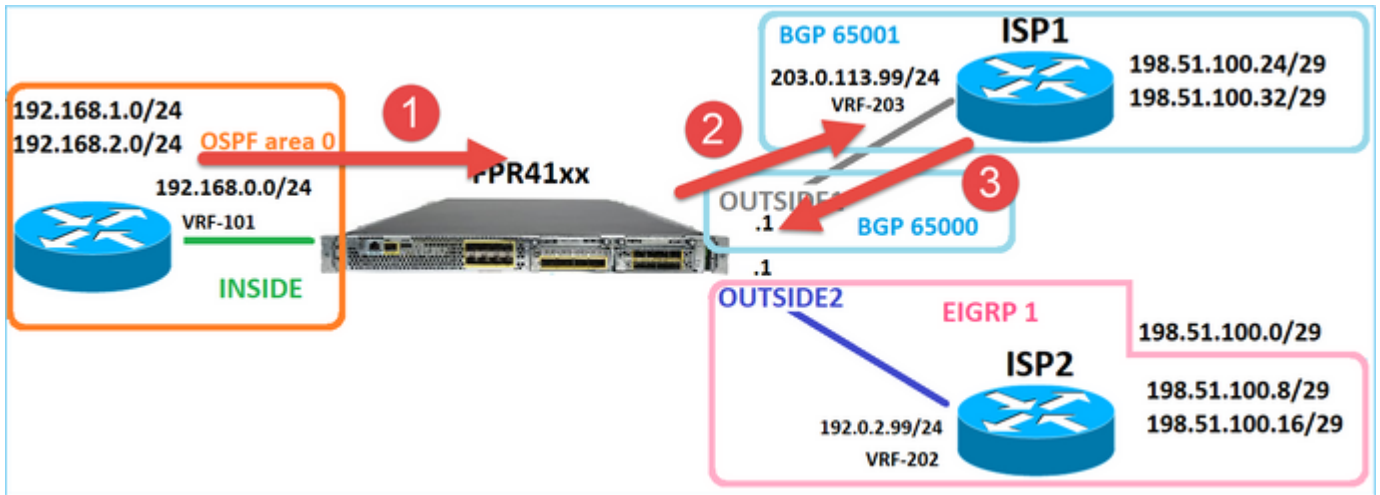
The packet details include the MAC address info, and a trace of the packets on OUTSIDE1 and OUTSIDE2 interfaces reveals the path of the packets:

```
firepower# show capture CAP01 detail
```

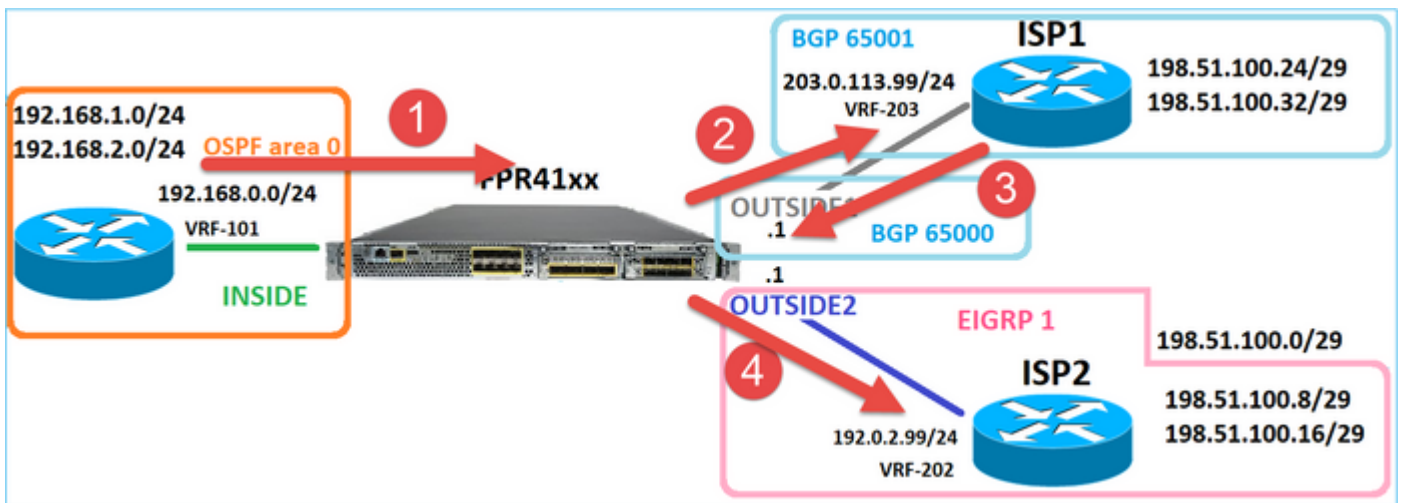
```
4 packets captured
```

```
1: 09:03:02.774358 00be.75f6.1dae 4c4e.35fc.fcd8 0x8100 Length: 62
802.1Q vlan#203 P0 192.168.1.1.32134 > 198.51.100.1.23: S [tcp sum ok] 3249840142:3249840142(0) win 4128
2: 09:03:02.774557 4c4e.35fc.fcd8 00be.75f6.1dae 0x8100 Length: 62
802.1Q vlan#203 P0 192.168.1.1.32134 > 198.51.100.1.23: S [tcp sum ok] 3249840142:3249840142(0) win 4128
3: 09:03:05.176702 00be.75f6.1dae 4c4e.35fc.fcd8 0x8100 Length: 62
802.1Q vlan#203 P0 192.168.1.1.32134 > 198.51.100.1.23: S [tcp sum ok] 3249840142:3249840142(0) win 4128
4: 09:03:05.176870 4c4e.35fc.fcd8 00be.75f6.1dae 0x8100 Length: 62
802.1Q vlan#203 P0 192.168.1.1.32134 > 198.51.100.1.23: S [tcp sum ok] 3249840142:3249840142(0) win 4128
```

```
4 packets shown
```

Trace of the packet that returns shows redirection to OUTSIDE2 interface due to Global Routing table Lookup:



```
firepower# show capture CAP01 packet-number 2 trace
```

```
4 packets captured
```

```
2: 09:03:02.774557 802.1Q vlan#203 P0 192.168.1.1.32134 > 198.51.100.1.23: S 3249840142:3249840142(0) wi
...
```

```
Phase: 3
```

```
Type: INPUT-ROUTE-LOOKUP
```

```
Subtype: Resolve Egress Interface
```

```
Result: ALLOW
```

```
Elapsed time: 7136 ns
```

```
Config:
```

```
Additional Information:
```

```
Found next-hop 192.0.2.99 using egress ifc OUTSIDE2(vrfid:0)
```

```
...
```

```
Phase: 10
```

```
Type: FLOW-CREATION
```

```
Subtype:
```

```
Result: ALLOW
```

```
Elapsed time: 12488 ns
```

Config:
Additional Information:
New flow created with id 13156, packet dispatched to next module

...

Phase: 13
Type: INPUT-ROUTE-LOOKUP-FROM-OUTPUT-ROUTE-LOOKUP
Subtype: Resolve Preferred Egress interface
Result: ALLOW
Elapsed time: 3568 ns
Config:
Additional Information:
Found next-hop 192.0.2.99 using egress ifc OUTSIDE2(vrfid:0)

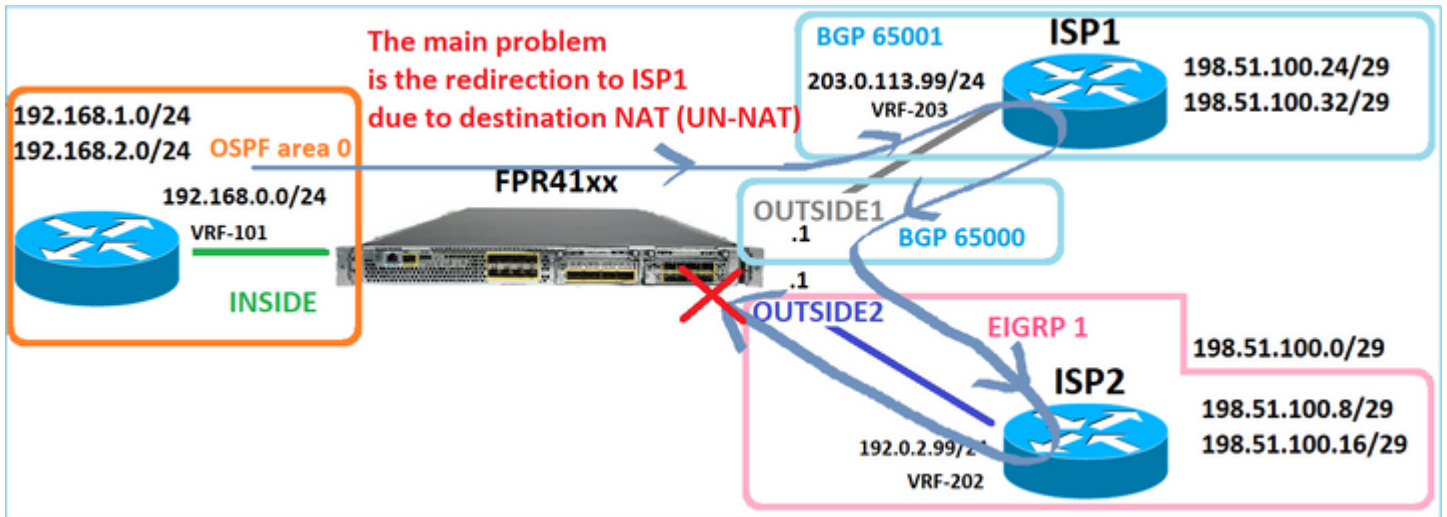
Phase: 14
Type: ADJACENCY-LOOKUP
Subtype: Resolve Nexthop IP address to MAC
Result: ALLOW
Elapsed time: 1338 ns
Config:
Additional Information:
Found adjacency entry for Next-hop 192.0.2.99 on interface OUTSIDE2
Adjacency :Active
MAC address 4c4e.35fc.fcd8 hits 0 reference 1

...

Result:
input-interface: OUTSIDE1(vrfid:0)
input-status: up
input-line-status: up
output-interface: OUTSIDE2(vrfid:0)
output-status: up
output-line-status: up
Action: allow
Time Taken: 111946 ns

1 packet shown
firepower#

The ISP2 router sends the reply (SYN/ACK), but this packet is redirected to ISP1 because it matches the established connection. The packet is dropped by the FTD due to no L2 adjacency in the ASP out table:



```
firepower# show capture CAP02 packet-number 2 trace
```

```
5 packets captured
```

```
2: 09:03:02.775457 802.1Q vlan#202 P0 198.51.100.1.23 > 192.168.1.1.32134: S 4075003210:4075003210(0) a
...
```

```
Phase: 3
```

```
Type: FLOW-LOOKUP
```

```
Subtype:
```

```
Result: ALLOW
```

```
Elapsed time: 2230 ns
```

```
Config:
```

```
Additional Information:
```

```
Found flow with id 13156, using existing flow
```

```
...
```

```
Phase: 7
```

```
Type: SUBOPTIMAL-LOOKUP
```

```
Subtype: suboptimal next-hop
```

```
Result: ALLOW
```

```
Elapsed time: 0 ns
```

```
Config:
```

```
Additional Information:
```

```
Input route lookup returned ifc INSIDE is not same as existing ifc OUTSIDE1
```

```
Result:
```

```
input-interface: OUTSIDE2(vrfid:0)
```

```
input-status: up
```

```
input-line-status: up
```

```
output-interface: INSIDE(vrfid:0)
```

```
output-status: up
```

```
output-line-status: up
```

```
Action: drop
```

```
Time Taken: 52628 ns
```

```
Drop-reason: (no-adjacency) No valid adjacency, Drop-location: frame 0x00005577204a7287 flow (NA)/NA
```

Case 3 – Forwarding based on Policy Based Routing (PBR)

After the connection flow lookup and the destination NAT lookup, PBR is the next item that can influence the egress interface determination. PBR is documented in: [Policy Based Routing](#)

For the PBR configuration on FMC, it is important to be aware of this guideline:
FlexConfig was used to configure PBR in FMC for FTD versions earlier than 7.1. You can still use FlexConfig to configure PBR in all versions. However, for an ingress interface, you cannot configure PBR using both FlexConfig and FMC's Policy Based Routing page.

In this case study, the FTD has a route towards 198.51.100.0/24 that points towards ISP2:

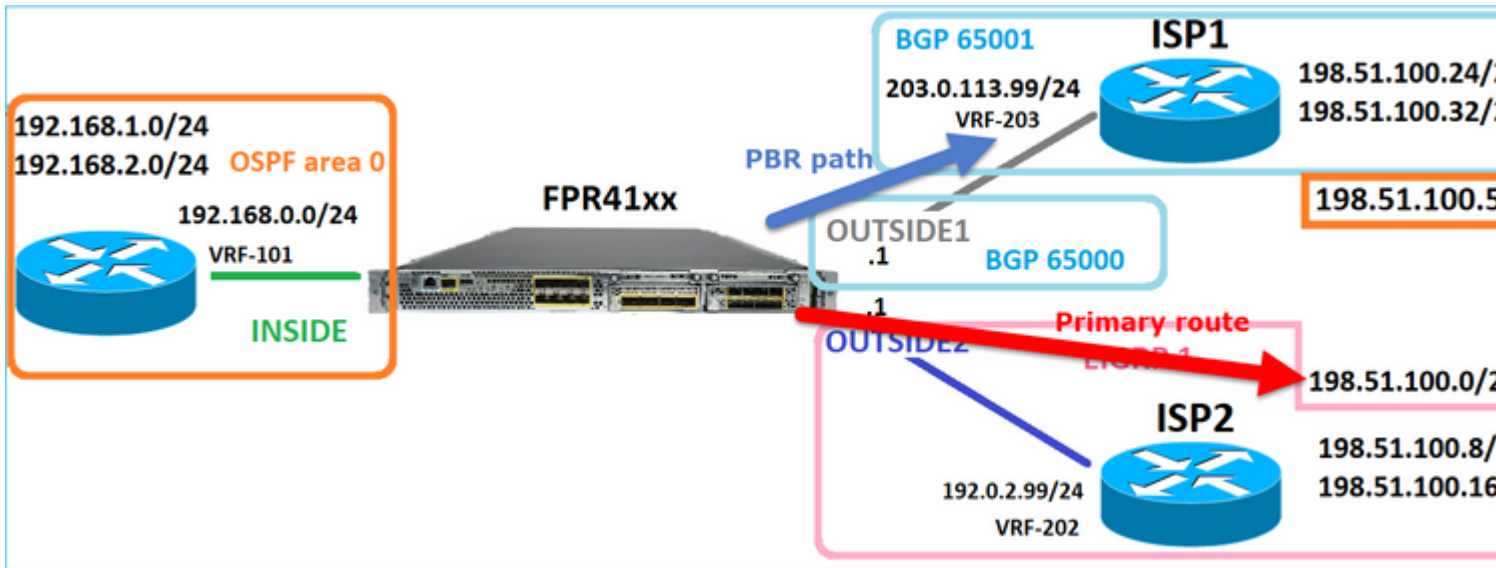
```
firepower# show route | begin Gate
Gateway of last resort is not set

C 192.0.2.0 255.255.255.0 is directly connected, OUTSIDE2
L 192.0.2.1 255.255.255.255 is directly connected, OUTSIDE2
C 192.168.0.0 255.255.255.0 is directly connected, INSIDE
L 192.168.0.1 255.255.255.255 is directly connected, INSIDE
O 192.168.1.1 255.255.255.255 [110/11] via 192.168.0.99, 5d01h, INSIDE
O 192.168.2.1 255.255.255.255 [110/11] via 192.168.0.99, 5d01h, INSIDE
S 198.51.100.0 255.255.255.248 [1/0] via 192.0.2.99, OUTSIDE2
D 198.51.100.8 255.255.255.248
[90/130816] via 192.0.2.99, 5d01h, OUTSIDE2
D 198.51.100.16 255.255.255.248
[90/130816] via 192.0.2.99, 5d01h, OUTSIDE2
B 198.51.100.24 255.255.255.248 [20/0] via 203.0.113.99, 5d00h
B 198.51.100.32 255.255.255.248 [20/0] via 203.0.113.99, 5d00h
C 203.0.113.0 255.255.255.0 is directly connected, OUTSIDE1
L 203.0.113.1 255.255.255.255 is directly connected, OUTSIDE1
```

Requirement

Configure a PBR Policy with these characteristics:

- Traffic from IP 192.168.2.0/24 destined to 198.51.100.5 must be sent to ISP1 (next-hop 203.0.113.99) while other sources must use the OUTSIDE2 interface.



Solution

In pre-7.1 releases, to configure PBR:

1. Create an Extended ACL that matches the interesting traffic (for example, PBR_ACL).
2. Create a route-map that matches the ACL created in Step 1, and set the desired next hop.
3. Create a FlexConfig Object that enables PBR on the ingress interface using the route map created in Step 2.

In post-7.1 releases, you can configure PBR using the pre-7.1 way, or you can use the new Policy Based Routing option under the Device > Routing section:

1. Create an Extended ACL that matches the interesting traffic (for example, PBR_ACL).
2. Add a PBR policy and specify:
 - a. The matching traffic
 - b. The ingress interface
 - c. The next-hop

Configure PBR (new way)

Step 1 – Define an Access List for the matching traffic.

Firewall Management Center
Objects / Object Management

Overview Analysis Policies Devices **Objects** Integration

> AAA Server
 > Access List **2**
 Extended
 Standard
 > Address Pools
 Application Filters
 AS Path
 Cipher Suite List
 > Community List
 > Distinguished Name
 DNS Server Group
 > External Attributes
 File List

Extended
 An access list object, also known as an access control list (ACL), selects the traffic to which a service will apply. Standard-Identifies t
 Supports IPv4 a

Edit Extended Access List Object

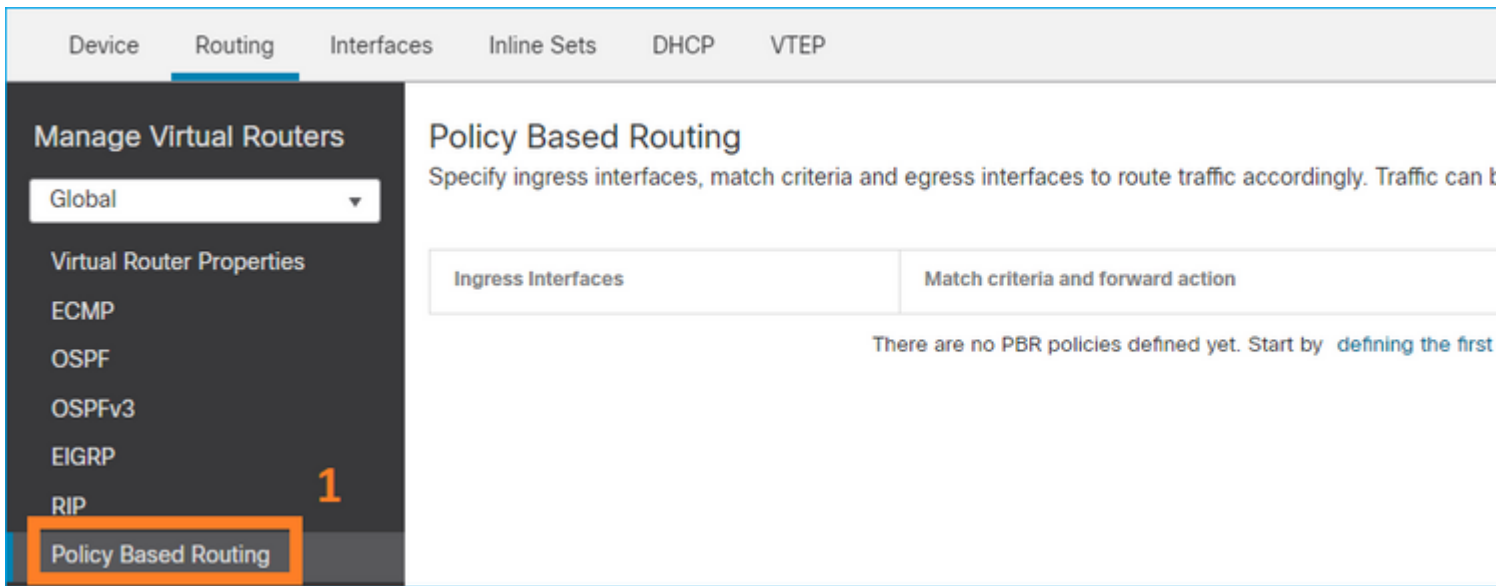
Name
 Name
 ACL_PBR

Entries (1)

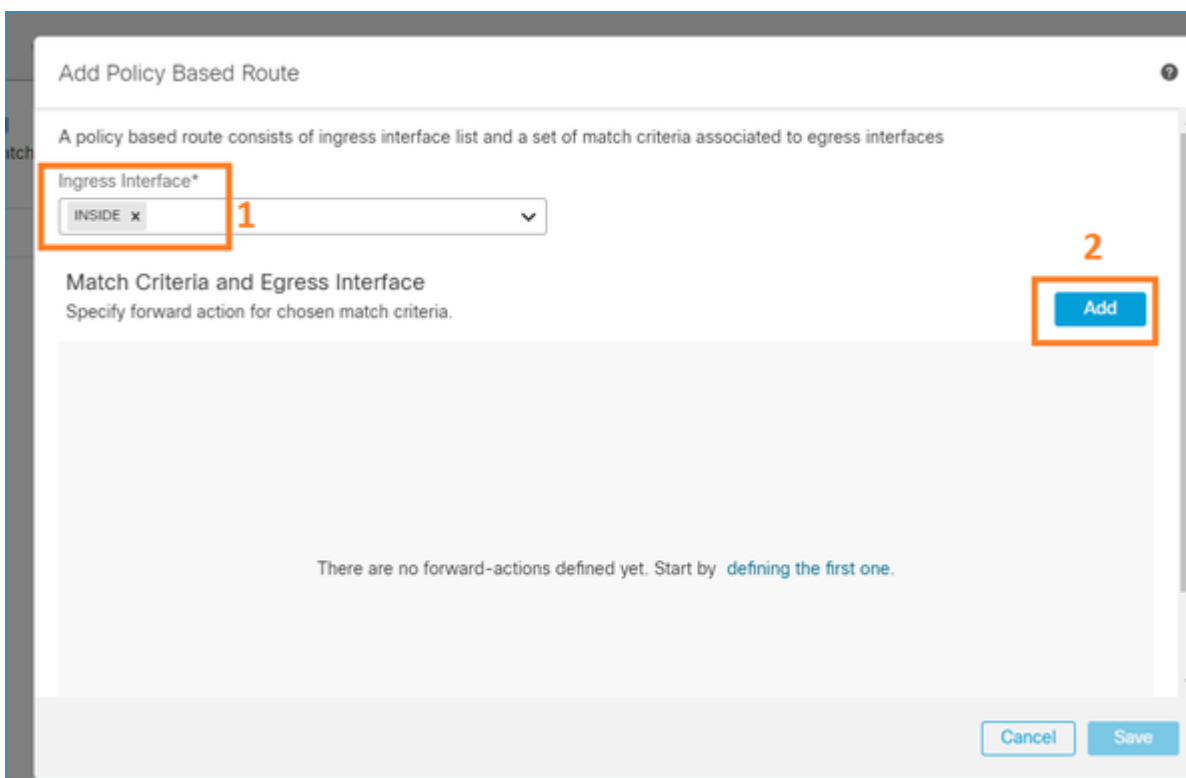
Sequence	Action	Source	Source Port	Destination	Destinat
1	Allow	192.168.2.0/24	Any	198.51.100.5	Any

Step 2 – Add a PBR Policy

Navigate to Devices > Device Management and edit the FTD device. Choose Routing > Policy Based Routing, and on the Policy Based Routing page, select Add.



Specify the ingress interface:



Specify the forwarding actions:

Add Forwarding Actions

Match ACL:* 1

Send To:* 2

IPv4 Addresses 3

IPv6 Addresses

Save and Deploy

Note: If you want to configure multiple egress interfaces you have to set in the 'Send To' field the 'Egress Interfaces' option (available as from version 7.0+). For more details check: [Configuration Example for Policy Based Routing](#)

Configure PBR (legacy way)

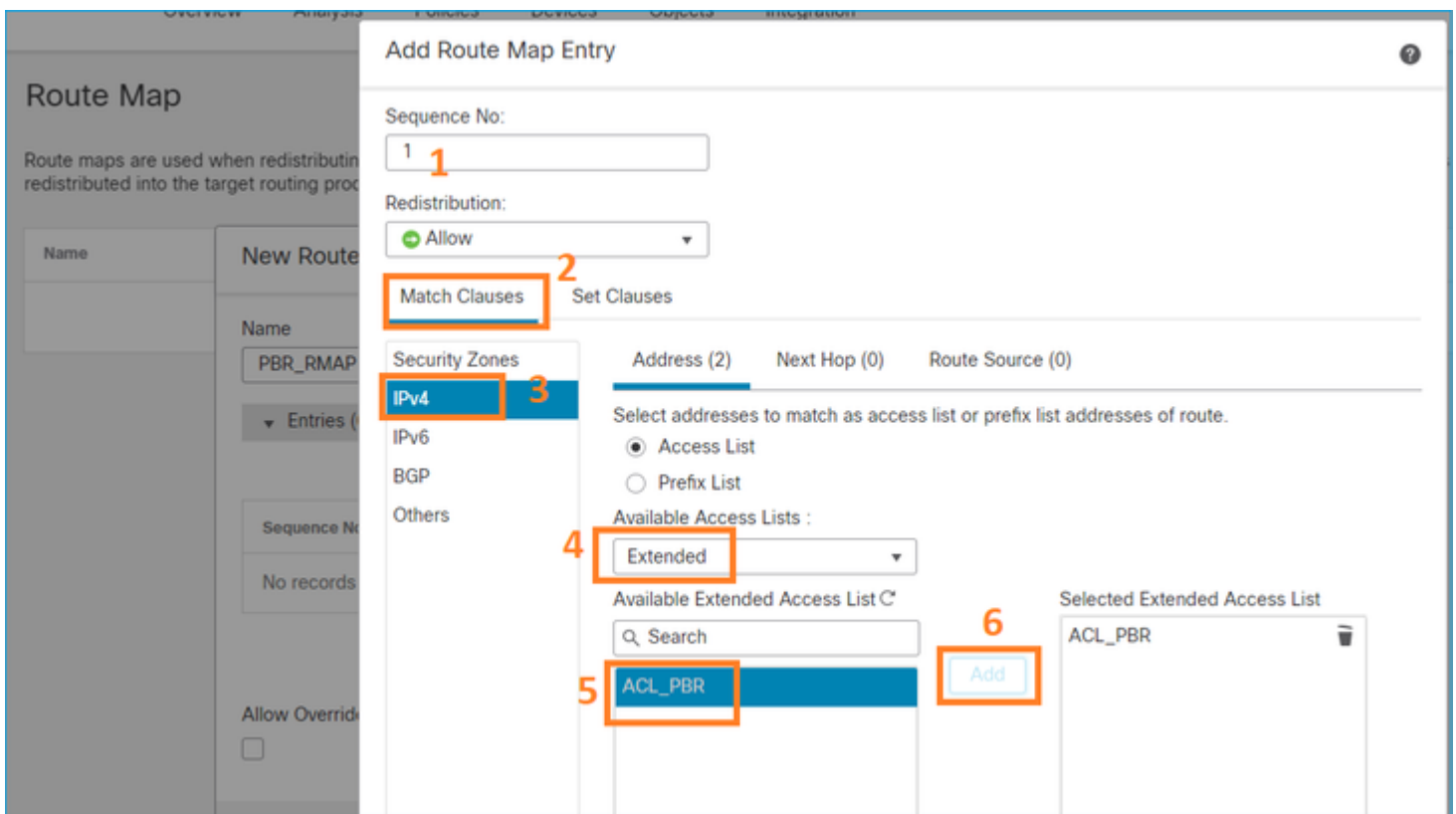
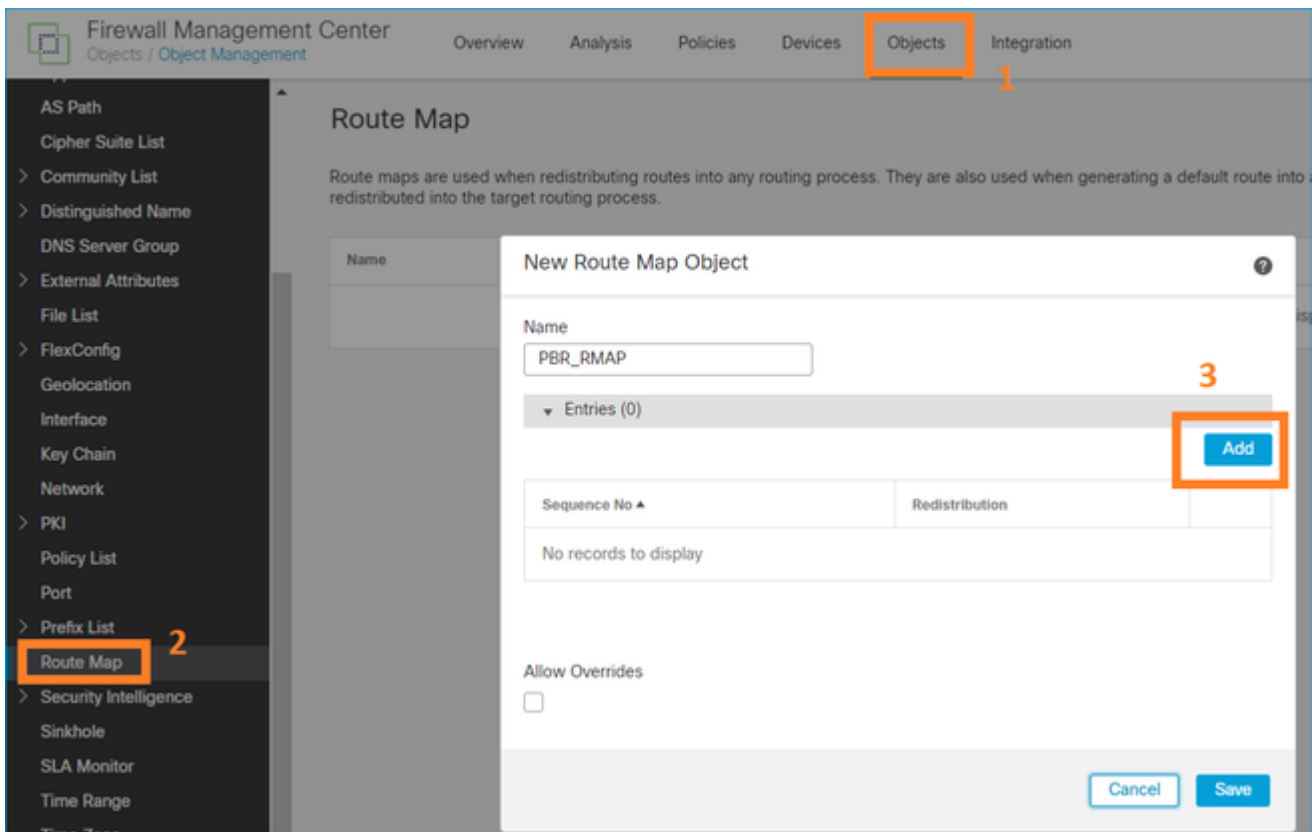
Step 1 – Define an Access List for the matching traffic.

The screenshot shows the Firewall Management Center interface. The 'Objects' tab is selected in the top navigation bar. In the left sidebar, 'Access List' is expanded, and 'Extended' is selected. The main content area shows the 'Edit Extended Access List Object' form. The 'Name' field is set to 'ACL_PBR'. Below, a table lists the entries for the access list:

Sequence	Action	Source	Source Port	Destination	Destination Port
1	Allow	192.168.2.0/24	Any	198.51.100.5	Any

Step 2 – Define a Route-Map that matches the ACL and sets the Next Hop.

First, define the Match Clause:



Define the Set Clause:

Edit Route Map Entry

Sequence No:

Redistribution:

Match Clauses **Set Clauses** 1

Metric Values **BGP Clauses** 2

AS Path Community List **Others** 3

Local Preference :
Range: 1-4294967295

Set Weight :
Range: 0-65535

Origin:

Local IGP

Incomplete

IPv4 settings:

Next Hop:

4

Specific IP :
Use comma to separate multiple values

Prefix List:

IPv6 settings:

Add and Save.

Step 3 – Configure the FlexConfig PBR Object.

First, copy (duplicate) the existing PBR object:

Firewall Management Center

Objects / Object Management

Overview Analysis Policies Devices **Objects** Integration Deploy

FlexConfig Object Add FlexConfig O

FlexConfig Object include device configuration commands, variables, and scripting language instructions

Name	Domain
Policy_Based_Routing	Global
Policy_Based_Routing_Clear	Global

AS Path
Cipher Suite List
Community List
Distinguished Name
DNS Server Group
External Attributes
File List
FlexConfig 1
FlexConfig Object
Text Object
Geolocation

Specify the Object name and remove the predefined route-map object:

Add FlexConfig Object

Name: **1**
FTD4100_PBR **Specify a new name**

Description:
The template is an example of PBR policy configuration. It

▲ Copy-pasting any rich text might introduce line breaks while generating CLI. Please verify the CLI before deployment.

Insert | Deployment: Once | Type: Append

```
interface Port-channel1.101  
policy-route route-map Sr-map-object 2 Specify the correct ingress interface  
3 Remove this route-map
```

Specify the new route-map:

Add FlexConfig Object

Name:
FTD4100_PBR

Description:
The template is an example of PBR policy configuration. It

▲ Copy-pasting any rich text might introduce line breaks while generating CLI. Please verify the CLI before deployment.

Insert **1** | Deployment: Once | Type: Append

- Insert Policy Object ▶ Text Object
- Insert System Variable ▶ Network
- Insert Secret Key
- Security Zones
- Standard ACL Object
- Extended ACL Object
- 2** Route Map

Insert Route Map Variable

Variable Name:
PBR_RMAP **1**

Description:

Available Objects **2**
Search
PBR_RMAP **3**

Selected Object
PBR_RMAP

Add

This is the end result:

Add FlexConfig Object

Name:

Description:

▲ Copy-pasting any rich text might introduce line breaks while generating CLI. Please verify the CLI before deployment.

Insert | | Deployment: | Type:

```
interface Port-channel1.101
  policy-route route-map $PBR_RMAP
```

Step 4 – Add the PBR Object to the FTD FlexConfig policy.

Firewall Management Center
Devices / Flexconfig Policy Editor

Overview Analysis Policies Devices Objects Integration Deploy

FTD4100_FlexConfig

Enter Description

Available FlexConfig FlexConfig Object

- User Defined **1**
 - FTD4100_PBR** **2**
 - no_ICMP
- System Defined
 - Default_DNS_Configure
 - Default_Inspection_Protocol_Disable
 - Default_Inspection_Protocol_Enable
 - DHCPv6_Prefix_Delegation_Configure
 - DHCPv6_Prefix_Delegation_UnConfigure

Selected Prepend FlexConfigs

#	Name	Description
---	------	-------------

Selected Append FlexConfigs

#	Name	Description
1	FTD4100_PBR	The template is an example of PBR p

Save and select Preview Config:

Preview FlexConfig

Select Device:

mzafeiro_FTD4100-1

```
route-map PBR_RMAP permit 1
match ip address ACL_PBR
set ip next-hop 203.0.113.99
vpn-addr-assign local
```

```
!INTERFACE_START
no logging FMC MANAGER_VPN_EVENT_LIST
```

```
!INTERFACE_END
```

```
###Flex-config Appended CLI ###
```

```
interface Port-channel1.101
 policy-route route-map PBR_RMAP
```

Finally, Deploy the policy.

Note: PBR cannot be configured using FlexConfig and FMC UI for the same ingress interface.

For PBR SLA configuration, check this document: [Configure PBR with IP SLAs for DUAL ISP on FTD Managed by FMC](#)

PBR Verification

Ingress interface verification:

```
firepower# show run interface Po1.101
!
interface Port-channel1.101
vlan 101
nameif INSIDE
cts manual
propagate sgt preserve-untag
policy static sgt disabled trusted
security-level 0
ip address 192.168.0.1 255.255.255.0
policy-route route-map FMC_GENERATED_PBR_1649228271478
ospf authentication null
```

Route-map verification:

```
firepower# show run route-map
!
route-map FMC_GENERATED_PBR_1649228271478 permit 5
  match ip address ACL_PBR
  set ip next-hop 203.0.113.99
```

```
firepower# show route-map
route-map FMC_GENERATED_PBR_1649228271478, permit, sequence 5
Match clauses:
ip address (access-lists): ACL_PBR

Set clauses:
adaptive-interface cost OUTSIDE1 (0)
```

Policy-route verification:

```
firepower# show policy-route
Interface Route map
Port-channel1.101 FMC_GENERATED_PBR_1649228271478
```

Packet-Tracer before and after the change:

Without PBR	With PBR
<pre>firepower# packet-tracer input INSIDE tcp 192.168.2.100 1111 198.51.100.5 23 Phase: 3 Type: INPUT-ROUTE-LOOKUP Subtype: Resolve Egress Interface Result: ALLOW Elapsed time: 11596 ns Config: Additional Information: Found next-hop 192.0.2.99 using egress ifc OUTSIDE2(vrfid:0) ... Phase: 13 Type: INPUT-ROUTE-LOOKUP-FROM-OUTPUT-ROUTE-LOOKUP Subtype: Resolve Preferred Egress interface Result: ALLOW Elapsed time: 6244 ns Config:</pre>	<pre>firepower# packet-tracer i ... Phase: 3 Type: SUBOPTIMAL-LOOKUP Subtype: suboptimal next-h Result: ALLOW Elapsed time: 39694 ns Config: Additional Information: Input route lookup returne Phase: 4 Type: ECMP load balancing Subtype: Result: ALLOW Elapsed time: 2230 ns Config: Additional Information: ECMP load balancing Found next-hop 203.0.113.9 Phase: 5 Type: PBR-LOOKUP Subtype: policy-route Result: ALLOW Elapsed time: 446 ns</pre>

```

Additional Information:
Found next-hop 192.0.2.99 using egress ifc OUTSIDE2(vrfid:0)

Phase: 14
Type: ADJACENCY-LOOKUP
Subtype: Resolve Nexthop IP address to MAC
Result: ALLOW
Elapsed time: 2230 ns
Config:
Additional Information:
Found adjacency entry for Next-hop 192.0.2.99 on interface OUTSIDE2
Adjacency :Active
MAC address 4c4e.35fc.fcd8 hits 0 reference 1

Result:
input-interface: INSIDE(vrfid:0)
input-status: up
input-line-status: up
output-interface: OUTSIDE2(vrfid:0)
output-status: up
output-line-status: up
Action: allow
Time Taken: 272058 ns

```

```

Config:
route-map FMC_GENERATED_PB
match ip address ACL_PBR
set adaptive-interface cos
Additional Information:
Matched route-map FMC_GENE
Found next-hop 203.0.113.9
...

Phase: 15
Type: ADJACENCY-LOOKUP
Subtype: Resolve Nexthop I
Result: ALLOW
Elapsed time: 5352 ns
Config:
Additional Information:
Found adjacency entry for
Adjacency :Active
MAC address 4c4e.35fc.fcd8

Result:
input-interface: INSIDE(vr
input-status: up
input-line-status: up
output-interface: OUTSIDE1
output-status: up
output-line-status: up
Action: allow
Time Taken: 825100 ns

```

Test with Real Traffic

Configure packet capture with a trace:

```

firepower# capture CAPI trace interface INSIDE match ip host 192.168.2.1 host 198.51.100.5
firepower# capture CAP01 trace interface OUTSIDE1 match ip host 192.168.2.1 host 198.51.100.5
firepower# capture CAP02 trace interface OUTSIDE2 match ip host 192.168.2.1 host 198.51.100.5

```

```

Router1# telnet 198.51.100.5 /vrf VRF-101 /source-interface lo2
Trying 198.51.100.5 ... Open

```

The capture shows:

```

firepower# show capture
capture CAPI type raw-data trace interface INSIDE [Capturing - 4389 bytes]
match ip host 192.168.2.1 host 198.51.100.5
capture CAP01 type raw-data trace interface OUTSIDE1 [Capturing - 4389 bytes]
match ip host 192.168.2.1 host 198.51.100.5
capture CAP02 type raw-data trace interface OUTSIDE2 [Capturing - 0 bytes]
match ip host 192.168.2.1 host 198.51.100.5

```

Trace of the TCP SYN packet:

```
firepower# show capture CAPI packet-number 1 trace
```

```
44 packets captured
```

```
1: 13:26:38.485585 802.1Q vlan#101 P0 192.168.2.1.49032 > 198.51.100.5.23: S 571152066:571152066(0) win  
...
```

```
Phase: 3
```

```
Type: SUBOPTIMAL-LOOKUP
```

```
Subtype: suboptimal next-hop
```

```
Result: ALLOW
```

```
Elapsed time: 13826 ns
```

```
Config:
```

```
Additional Information:
```

```
Input route lookup returned ifc OUTSIDE2 is not same as existing ifc OUTSIDE1
```

```
Phase: 4
```

```
Type: ECMP load balancing
```

```
Subtype:
```

```
Result: ALLOW
```

```
Elapsed time: 1784 ns
```

```
Config:
```

```
Additional Information:
```

```
ECMP load balancing
```

```
Found next-hop 203.0.113.99 using egress ifc OUTSIDE1(vrfid:0)
```

```
Phase: 5
```

```
Type: PBR-LOOKUP
```

```
Subtype: policy-route
```

```
Result: ALLOW
```

```
Elapsed time: 446 ns
```

```
Config:
```

```
route-map FMC_GENERATED_PBR_1649228271478 permit 5
```

```
match ip address ACL_PBR
```

```
set adaptive-interface cost OUTSIDE1
```

```
Additional Information:
```

```
Matched route-map FMC_GENERATED_PBR_1649228271478, sequence 5, permit
```

```
Found next-hop 203.0.113.99 using egress ifc OUTSIDE1
```

```
...
```

```
Phase: 15
```

```
Type: ADJACENCY-LOOKUP
```

```
Subtype: Resolve Nexthop IP address to MAC
```

```
Result: ALLOW
```

```
Elapsed time: 4906 ns
```

```
Config:
```

```
Additional Information:
```

```
Found adjacency entry for Next-hop 203.0.113.99 on interface OUTSIDE1
```

```
Adjacency :Active
```

```
MAC address 4c4e.35fc.fcd8 hits 348 reference 2
```

```
...
```

```
Result:
```

```
input-interface: INSIDE(vrfid:0)
```

```
input-status: up
```

```
input-line-status: up
output-interface: OUTSIDE1(vrfid:0)
output-status: up
output-line-status: up
Action: allow
Time Taken: 222106 ns
```

The ASP PBR table shows the policy hit counts:

```
firepower# show asp table classify domain pbr
```

Input Table

```
in id=0x1505f26d3420, priority=2147483642, domain=pbr, deny=false
hits=7, user_data=0x1505f26e7590, cs_id=0x0, use_real_addr, flags=0x0, protocol=0
src ip/id=192.168.2.0, mask=255.255.255.0, port=0, tag=any
dst ip/id=198.51.100.5, mask=255.255.255.255, port=0, tag=any, dscp=0x0, nsg_id=none
input_ifc=INSIDE(vrfid:0), output_ifc=any
```

Output Table:

L2 - Output Table:

L2 - Input Table:

Last clearing of hits counters: Never

Note: The packet-tracer also increases the hit counter.

PBR Debug

Warning: In a production environment, the debug can produce a lot of messages.

Enable this debug:

```
firepower# debug policy-route
debug policy-route enabled at level 1
```

Send real traffic:

```
Router1# telnet 198.51.100.5 /vrf VRF-101 /source-interface lo2
Trying 198.51.100.5 ... Open
```

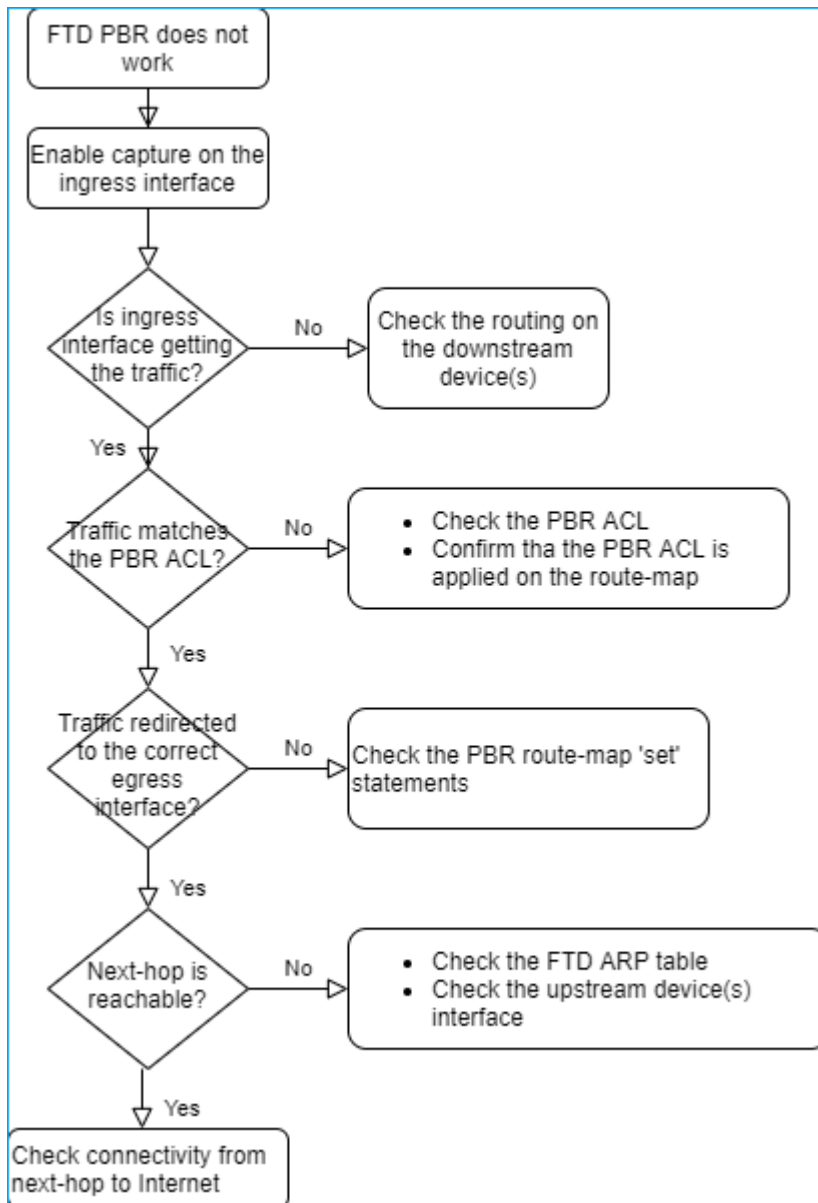
The debug shows:

firepower#

pbr: policy based route lookup called for 192.168.2.1/32 to 198.51.100.5/23 proto 6 sub_proto 0 received
pbr: First matching rule from ACL(2)
pbr: route map FMC_GENERATED_PBR_1649228271478, sequence 5, permit; proceed with policy routing
pbr: policy based routing applied; egress_ifc = OUTSIDE1 : next_hop = 203.0.113.99

Note: Packet-tracer also generates a debug output.

This flowchart can be used to troubleshoot PBR:



Summary of PBR commands

To verify the configuration:

```
show run route-map  
show run interface
```

In case SLA Monitor is also used with PBR:

```
show run sla monitor
show run track
```

To verify the operation:

```
show route-map
packet-tracer
capture w/trace (for example, capture CAPI interface INSIDE trace match ip host 192.168.0.1 host 203.0.113.1)
ASP drop capture (for example, capture ASP type asp-drop all)
show asp table classify domain pbr
show log
show arp
```

In case SLA Monitor is also used with PBR:

```
show sla monitor operational-state
show sla monitor configuration
show track
```

To debug PBR:

```
debug policy-route
show asp drop
```

Case 4 – Forwarding based on Global Routing Lookup

After the connection lookup, NAT lookup, and PBR, the last item that is checked to determine the egress interface is the Global Routing table.

Routing Table Verification

Let us examine an FTD routing table output:

```

firepower# show route

Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, V - VPN
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, + - replicated route
       SI - Static InterVRF, BI - BGP InterVRF

Gateway of last resort is not set

C      192.0.2.0 255.255.255.0 is directly connected, OUTSIDE2
L      192.0.2.1 255.255.255.255 is directly connected, OUTSIDE2
C      192.168.0.0 255.255.255.0 is directly connected, INSIDE
L      192.168.0.1 255.255.255.255 is directly connected, INSIDE
O      192.168.1.1 255.255.255.255
O      192.168.2.1 255.255.255.255
O      [110/11] via 192.168.0.99, 01:36:53, INSIDE
O      192.168.2.1 255.255.255.255
O      [110/11] via 192.168.0.99, 01:36:53, INSIDE
S      198.51.100.0 255.255.255.248 [1/0] via 192.0.2.99, OUTSIDE2
D      198.51.100.8 255.255.255.248
D      [90/128512] via 192.0.2.99, 15:13:23, OUTSIDE2
D      198.51.100.16 255.255.255.248
D      [90/128512] via 192.0.2.99, 15:13:23, OUTSIDE2
B      198.51.100.24 255.255.255.248 [20/0] via 203.0.113.99, 15:13:26
B      198.51.100.32 255.255.255.248 [20/0] via 203.0.113.99, 15:13:26

```

The main goal of the routing process is to find the next hop. The route selection is in this order:

1. Longest match wins
2. Lowest AD (between different routing protocol sources)
3. Lowest Metric (in case routes are learned from the same source - routing protocol)

How the routing table is populated:

- IGP (R, D, EX, O, IA, N1, N2, E1, E2, i, su, L1, L2, ia, o)
- BGP (B)
- BGP InterVRF (BI)
- Static (S)
- Static InterVRF (SI)
- Connected (C)
- local IPs (L)
- VPN (V)
- Redistribution
- Default

To view the routing table summary use this command:

<#root>

```
firepower#
```

```
show route summary
```

```
IP routing table maximum-paths is 8
```

Route Source	Networks	Subnets	Replicates	Overhead	Memory (bytes)
connected	0	8	0	704	2368
static	0	1	0	88	296
ospf 1	0	2	0	176	600
Intra-area: 2 Inter-area: 0 External-1: 0 External-2: 0					
NSSA External-1: 0 NSSA External-2: 0					
bgp 65000	0	2	0	176	592
External: 2 Internal: 0 Local: 0					
eigrp 1	0	2	0	216	592
internal	7				3112
Total	7	15	0	1360	7560

You can track the routing table updates with this command:

```
<#root>
```

```
firepower#
```

```
debug ip routing
```

```
IP routing debugging is on
```

For example, this is what the debug shows when OSPF route 192.168.1.0/24 is removed from the global routing table:

```
<#root>
```

```
firepower#
```

```
RT: ip_route_delete 192.168.1.0 255.255.255.0 via 192.0.2.99, INSIDE
```

```
ha_cluster_synced 0 routetype 0
```

```
RT: del 192.168.1.0 via 192.0.2.99, ospf metric [110/11]NP-route: Delete-Output 192.168.1.0/24 hop_count:1
```

```
RT: delete network route to 192.168.1.0 255.255.255.0NP-route: Delete-Output 192.168.1.0/24 hop_count:1
```

```
NP-route: Delete-Input 192.168.1.0/24 hop_count:1 Distance:110 Flags:0X0 , via 0.0.0.0, INSIDE
```

When it is added back:

```
<#root>
```

```
firepower#
```

```
RT: NP-route: Add-Output 192.168.1.0/24 hop_count:1 , via 192.0.2.99, INSIDE
```

NP-route: Add-Input 192.168.1.0/24 hop_count:1 Distance:110 Flags:0X0 , via 192.0.2.99, INSIDE

Null0 Interface

Null0 interface can be used to drop unwanted traffic. This drop has less performance impact than the drop in the traffic with an Access Control Policy (ACL) rule.

Requirement

Configure a Null0 route for 198.51.100.4/32 host.

Solution

The screenshot shows the Cisco Firepower 4140 Threat Defense configuration interface for device FTD4100-1. The 'Routing' tab is active, and the 'Manage Virtual Routers' sidebar is open, with 'Static Route' selected (marked with a red '1'). The main area displays a table of IPv4 routes:

Network	Interface
IPv4 Routes	
net_198.51.100.0_29bits	OUTSIDE1
net_198.51.100.0_29bits	OUTSIDE2
IPv6 Routes	

An 'Add Static Route Configuration' dialog is open on the right. The 'Type' is set to IPv4. The 'Interface*' dropdown is set to 'Null0' (marked with a red '2'). The 'Available Network' search box contains 'host_198.51.100.4', and the selected network 'host_198.51.100.4' is highlighted (marked with a red '3'). The 'Gateway*' field is empty.

Save and Deploy.

Verification:

```
<#root>
```

```
firepower#
```

```
show run route
```

```
route OUTSIDE2 198.51.100.0 255.255.255.248 192.0.2.99 1
route OUTSIDE1 198.51.100.0 255.255.255.248 203.0.113.99 200
```

```
route Null0 198.51.100.4 255.255.255.255 1
```

```
<#root>
```

```
firepower#
```

```
show route | include 198.51.100.4
```

```
S 198.51.100.4 255.255.255.255 [1/0] is directly connected, Null0
```

Try to access the remote host:

```
<#root>
```

```
Router1#
```

```
ping vrf VRF-101 198.51.100.4
```

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 198.51.100.4, timeout is 2 seconds:

```
.....
```

```
Success rate is 0 percent (0/5)
```

The FTD logs show:

```
<#root>
```

```
firepower#
```

```
show log | include 198.51.100.4
```

```
Apr 12 2022 12:35:28:
```

```
%FTD-6-110002: Failed to locate egress interface for ICMP from INSIDE:192.168.0.99/0 to 198.51.100.4/0
```

ASP drops show:

```
<#root>
```

```
firepower#
```

```
show asp drop
```

Frame drop:

Equal Cost Multi-Path (ECMP)

Traffic Zones

- ECMP Traffic Zone allows a user to group interfaces together (referred to as an ECMP Zone).
- This allows ECMP routing as well as load-balancing of traffic across multiple interfaces.
- When interfaces are associated with ECMP Traffic Zone, the user is able to create Equal-Cost Static Routes across the interfaces. Equal-Cost Static Routes are routes to the same destination network with the same metric value.

Before version 7.1, Firepower Threat Defense supported ECMP routing through FlexConfig policies. As from the 7.1 release, you can group interfaces into traffic zones and configure ECMP routing in Firepower Management Center.

EMCP is documented in: [ECMP](#)

In this example, there is asymmetric routing, and return traffic is dropped:

```
<#root>
```

```
firepower#
```

```
show log
```

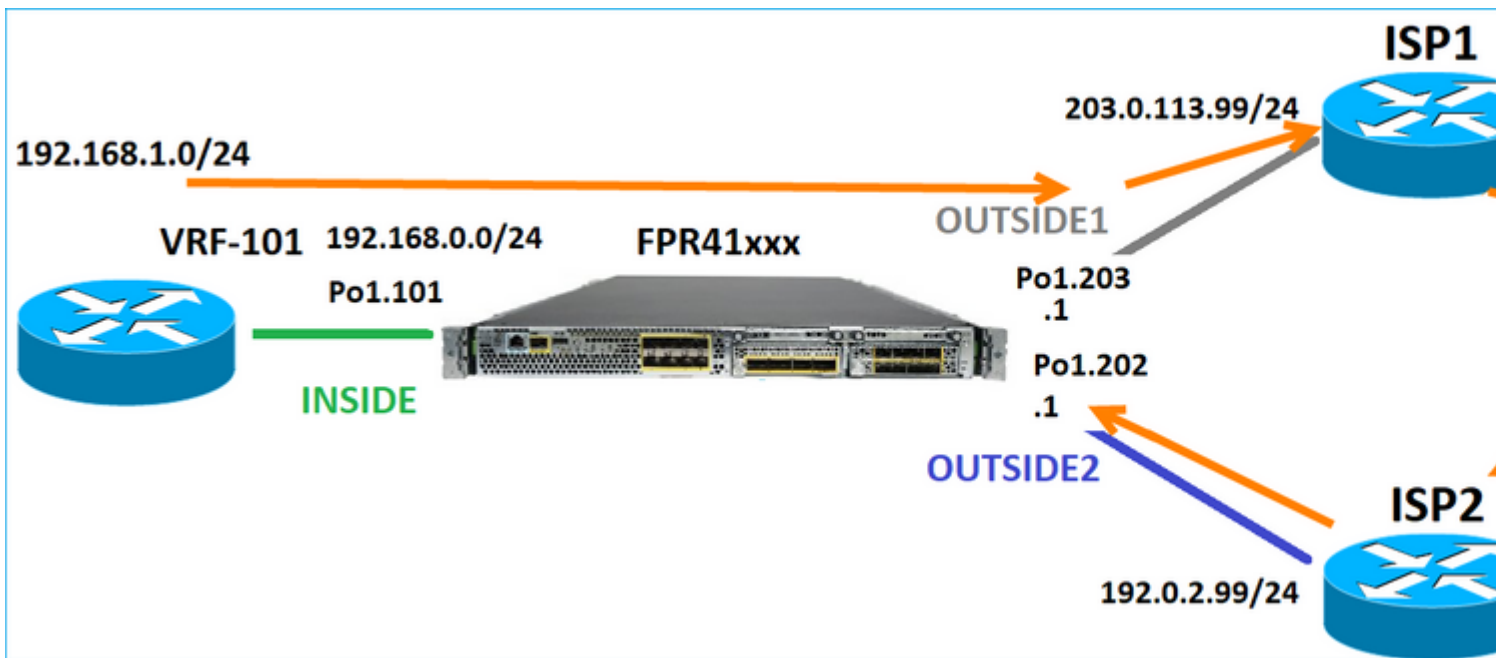
```
Apr 13 2022 07:20:48: %FTD-6-302013:
```

```
B
```

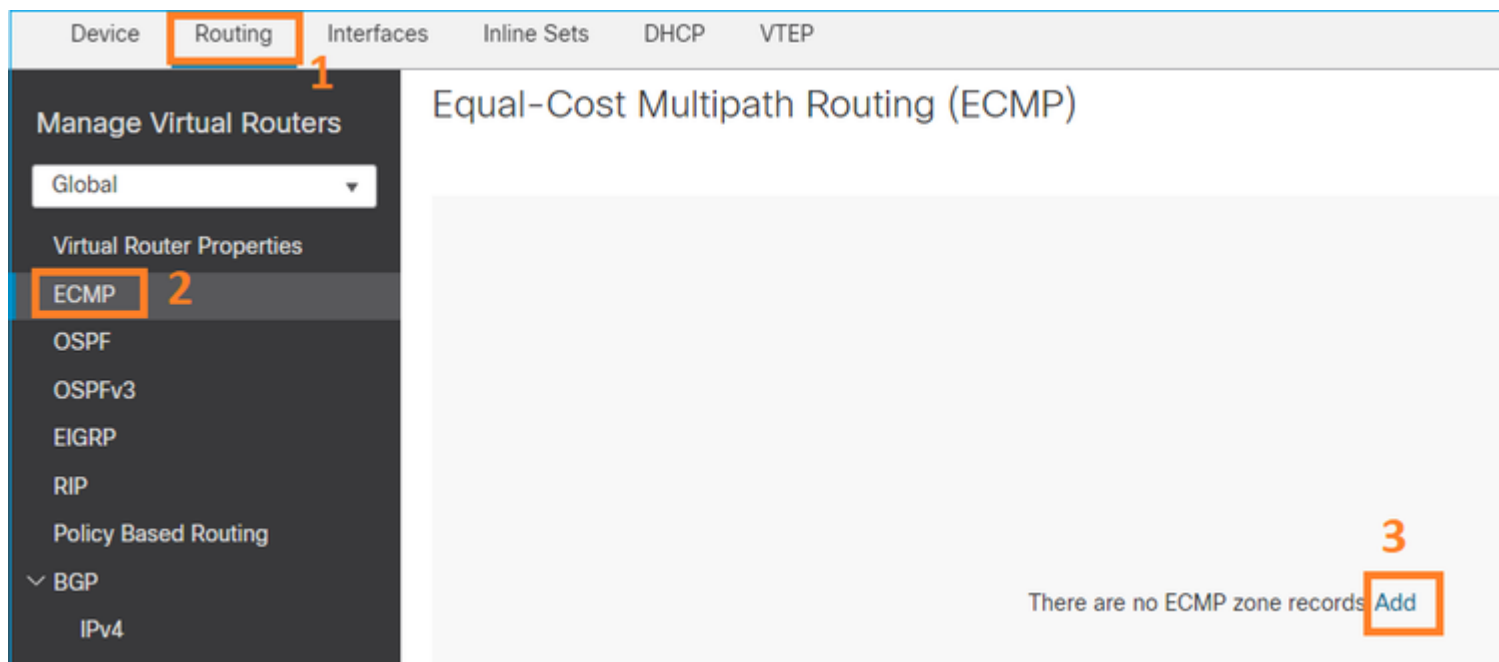
```
uilt inbound TCP connection 4046 for INSIDE:192.168.1.1/23943 (192.168.1.1/23943) to OUTSIDE1:198.51.100.100/23
```

```
Apr 13 2022 07:20:48: %FTD-6-106015:
```

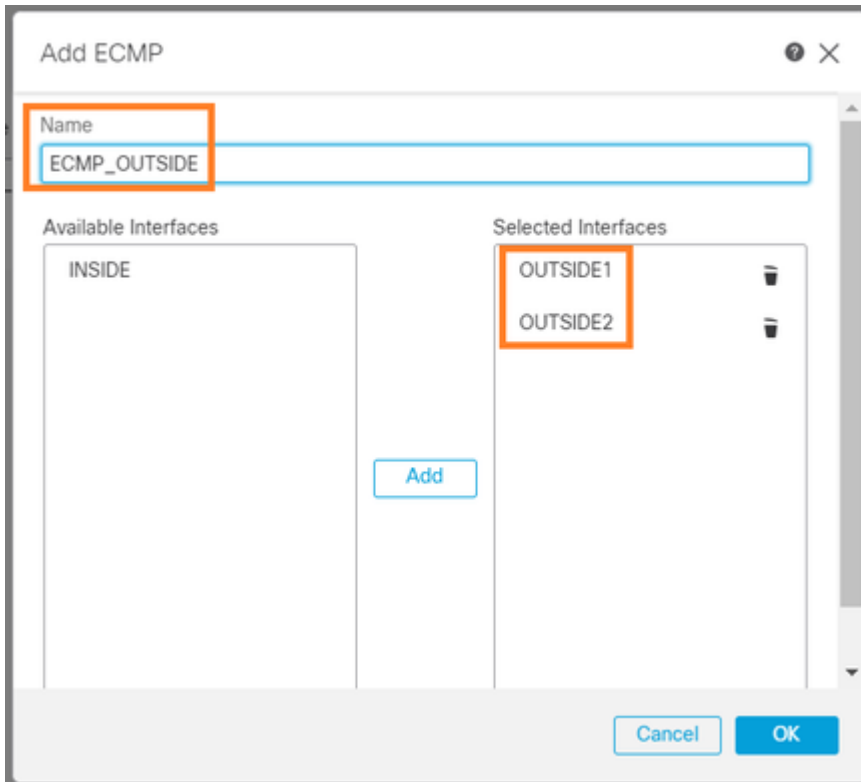
```
Deny TCP (no connection) from 198.51.100.100/23 to 192.168.1.1/23943 flags SYN ACK on interface OUTSIDE1
```



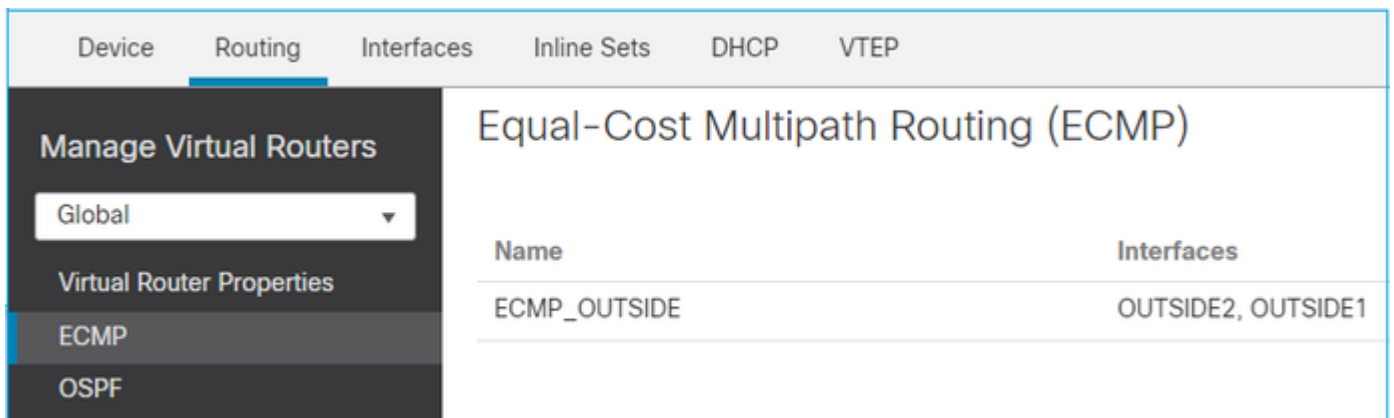
Configure ECMP from the FMC UI:



Add the 2 interfaces in the ECMP group:



The result:



Save and Deploy.

ECMP zone verification:

```
<#root>
```

```
firepower#
```

```
show run zone
```

```
zone ECMP_OUTSIDE ecmp
```

```
firepower#
```

```
show zone
```

Zone: ECMP_OUTSIDE ecmp

Security-level: 0

Zone member(s): 2

OUTSIDE1 Port-channel1.203

OUTSIDE2 Port-channel1.202

Interface verification:

<#root>

firepower#

show run int po1.202

```
!  
interface Port-channel1.202  
vlan 202  
nameif OUTSIDE2  
cts manual  
propagate sgt preserve-untag  
policy static sgt disabled trusted  
security-level 0
```

zone-member ECMP_OUTSIDE

ip address 192.0.2.1 255.255.255.0

firepower#

show run int po1.203

```
!  
interface Port-channel1.203  
vlan 203  
nameif OUTSIDE1  
cts manual  
propagate sgt preserve-untag  
policy static sgt disabled trusted  
security-level 0
```

zone-member ECMP_OUTSIDE

ip address 203.0.113.1 255.255.255.0

Now, the return traffic is allowed, and the connection is UP:

<#root>

Router1#

```
telnet 198.51.100.100 /vrf VRF-101 /source-interface lo1
```

Trying 198.51.100.100 ... Open

Capture on ISP1 interface shows the egress traffic:

<#root>

firepower#

```
show capture CAP1
```

5 packets captured

```
1: 10:03:52.620115 802.1Q vlan#203 P0 192.168.1.1.56199 > 198.51.100.100.23: S 1782458734:1782458734(0)
2: 10:03:52.621992 802.1Q vlan#203 P0 192.168.1.1.56199 > 198.51.100.100.23: . ack 2000807246 win 4128
3: 10:03:52.622114 802.1Q vlan#203 P0 192.168.1.1.56199 > 198.51.100.100.23: . ack 2000807246 win 4128
4: 10:03:52.622465 802.1Q vlan#203 P0 192.168.1.1.56199 > 198.51.100.100.23: P 1782458735:1782458753(18)
5: 10:03:52.622556 802.1Q vlan#203 P0 192.168.1.1.56199 > 198.51.100.100.23: . ack 2000807246 win 4128
```

Capture on ISP2 interface shows the return traffic:

<#root>

firepower#

```
show capture CAP2
```

6 packets captured

```
1: 10:03:52.621305 802.1Q vlan#202 P0 198.51.100.100.23 > 192.168.1.1.56199:
s
2000807245:2000807245(0)
ack
1782458735 win 64240 <mss 1460>
3: 10:03:52.623808 802.1Q vlan#202 P0 198.51.100.100.23 > 192.168.1.1.56199: . ack 1782458753 win 64222
```

FTD Management Plane

The FTD has 2 Management Planes:

- Management0 interface – Provides access to the Firepower subsystem
- LINA diagnostic interface – Provide access to FTD LINA subsystem

To configure and verify the Management0 interface, use the configure network and show network commands respectively.

On the other hand, the LINA interfaces provide access to the LINA itself. The FTD interface entries in the FTD RIB can be seen as Local routes:

```
<#root>
firepower#
show route | include L

Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
L 192.0.2.1 255.255.255.255 is directly connected, OUTSIDE2
L 192.168.0.1 255.255.255.255 is directly connected, INSIDE
L 203.0.113.1 255.255.255.255 is directly connected, OUTSIDE1
```

Similarly, they can be seen as identity entries in the ASP routing table:

```
<#root>
firepower#
show asp table routing | include identity

in 169.254.1.1 255.255.255.255 identity
in
192.0.2.1 255.255.255.255 identity

in
203.0.113.1 255.255.255.255 identity

in
192.168.0.1 255.255.255.255 identity

in ff02::1 ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff identity
in ff02::1:ff01:3 ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff identity
in ff02::1:ff00:1 ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff identity
in fe80::200:ff:fe01:3 ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff identity
in fd00:0:0:1::1 ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff identity
out 0.0.0.0 0.0.0.0 via 0.0.0.0, identity
out :: :: via 0.0.0.0, identity
```

Main Point

When a packet arrives on FTD, and the destination IP matches one of the identity IPs, the FTD knows that it has to consume the packet.

FTD LINA Diagnostic Interface Routing

FTD (like an ASA that runs post-9.5 code) maintains a VRF-like routing table for any interface that is configured as management-only. An example of such an interface is the diagnostic interface.

While FMC does not allow you (without ECMP) to configure 2 default routes on 2 different interfaces with the same metric, you can configure 1 default route on an FTD data interface and another default route on the diagnostic interface:

Network ▲	Interface	Leaked from Virtual Router	Gateway
▼ IPv4 Routes			
any-ipv4	diagnostic	Global	gw_10.62.148.1
any-ipv4	OUTSIDE1	Global	203.0.113.99

The data plane traffic uses the global table default gateway, while the management plane traffic uses the diagnostic default GW:

```
<#root>
```

```
firepower#
```

```
show route management-only
```

```
Routing Table: mgmt-only
```

```
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP  
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area  
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2  
E1 - OSPF external type 1, E2 - OSPF external type 2, V - VPN  
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2  
ia - IS-IS inter area, * - candidate default, U - per-user static route  
o - ODR, P - periodic downloaded static route, + - replicated route  
SI - Static InterVRF, BI - BGP InterVRF
```

```
Gateway of last resort is 10.62.148.1 to network 0.0.0.0
```

```
S* 0.0.0.0 0.0.0.0 [1/0] via 10.62.148.1, diagnostic
```

The global routing table gateway:

```
<#root>
```

```
firepower#
```

```
show route | include S\*|Gateway
```

```
Gateway of last resort is 203.0.113.99 to network 0.0.0.0
```

```
S* 0.0.0.0 0.0.0.0 [1/0] via 203.0.113.99, OUTSIDE1
```

When you send traffic from the FTD (from-the-box traffic), the egress interface is selected based on:

1. Global routing table
2. Management-only routing table

You can overwrite the egress interface selection if you manually specify the egress interface.

Try to ping the diagnostic interface gateway. If you do not specify the source interface, the ping fails because FTD first uses the global routing table which, in this case, it contains a default route. If there is no route in the global table, the FTD does a route lookup on the management-only routing table:

```
<#root>
```

```
firepower#
```

```
ping 10.62.148.1
```

```
Type escape sequence to abort.
```

```
Sending 5, 100-byte ICMP Echos to 10.62.148.1, timeout is 2 seconds:
```

```
?????
```

```
Success rate is 0 percent (0/5)
```

```
firepower#
```

```
show capture CAP1 | include 10.62.148.1
```

```
1: 10:31:22.970607 802.1Q vlan#203 P0
```

```
203.0.113.1 > 10.62.148.1 icmp: echo request
```

```
2: 10:31:22.971431 802.1Q vlan#203 P0
```

```
10.1.1.2 > 203.0.113.1 icmp: host 10.62.148.1 unreachable
```

```
<#root>
```

```
firepower#
```

```
ping diagnostic 10.62.148.1
```

```
Type escape sequence to abort.
```

Sending 5, 100-byte ICMP Echos to 10.62.148.1, timeout is 2 seconds:

!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms

The same applies if you try to copy a file from LINA CLI with the copy command.

Bidirectional Forwarding Detection (BFD)

BFD support was added on classic ASA version 9.6 and only for BGP protocol: [Bidirectional Forwarding Detection Routing](#)

On FTD:

- BGP IPv4 and BGP IPv6 protocols are supported (software 6.4).
- OSPFv2, OSPFv3, and EIGRP protocols are not supported.
- BFD for Static Routes is not supported.

Virtual Routers (VRF)

VRF support was added in the 6.6 release. For more details check this document: [Configuration Examples for Virtual Routers](#)

Related Information

- [FTD Static and Default Routes](#)