

# Clarify FTD Access Control Policy Rule Actions

## Contents

---

### [Introduction](#)

### [Prerequisites](#)

[Requirements](#)

[Components Used](#)

[Related Products](#)

### [Background Information](#)

[How ACP is Deployed](#)

### [Configure](#)

[ACP Available Actions](#)

[How ACP and Prefilter Policy Interact](#)

[ACP Block Action](#)

[Scenario 1. Early LINA Drop](#)

[Scenario 2. Drop Due to Snort Verdict](#)

[ACP Block with reset Action](#)

[ACP Allow Action](#)

[Scenario 1. ACP Allow Action \(L3/L4 Conditions\)](#)

[Scenario 2. ACP Allow Action \(L3-7 Conditions\)](#)

[Scenario 3. Snort Fast-Forward verdict with Allow](#)

[ACP Trust Action](#)

[Scenario 1. ACP Trust Action](#)

[Scenario 2. ACP Trust Action \(without SI, QoS, and Identity Policy\)](#)

[Prefilter Policy Block Action](#)

[Prefilter Policy Fastpath Action](#)

[Prefilter Policy Fastpath Action \(Inline-Set\)](#)

[Prefilter Policy Fastpath Action \(Inline-Set with Tap\)](#)

[Prefilter Policy Analyze Action](#)

[Scenario 1. Prefilter Analyze with ACP Block Rule](#)

[Scenario 2. Prefilter Analyze with ACP Allow Rule](#)

[Scenario 3. Prefilter Analyze with ACP Trust Rule](#)

[Scenario 4. Prefilter Analyze with ACP Trust Rule](#)

[ACP Monitor Action](#)

[ACP Interactive Block Action](#)

[ACP Interactive Block with Reset Action](#)

[FTD Secondary Connections and Pinholes](#)

[FTD Rule Guidelines](#)

### [Summary](#)

### [Related Information](#)

---

## Introduction

This document describes the various actions available on the Firepower Threat Defense (FTD) Access Control Policy (ACP) and Prefilter Policy.

## Prerequisites

### Requirements

Cisco recommends that you have knowledge of these topics:

- Flow Offload
- Packet captures on Firepower Threat Defense appliances
- Packet tracer and capture with trace option on FTD appliances

### Components Used

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

- Cisco Firepower 4110 Threat Defense Version 6.4.0 (Build 113) and 6.6.0 (Build 90)
- Firepower Management Center (FMC) Version 6.4.0 (Build 113) and 6.6.0 (Build 90)


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 be also used with these hardware and software versions:

- ASA5506-X, ASA5506W-X, ASA5506H-X, ASA5508-X, ASA5516-X
- ASA5512-X, ASA5515-X, ASA5525-X, ASA5545-X, ASA5555-X
- FPR1000, FPR2100, FPR4100, FPR9300
- VMware (ESXi), Amazon Web Services (AWS), Kernel-based Virtual Machine (KVM)
- Integrated Service Router (ISR) router module
- FTD software version 6.1.x and later

---

 **Note:** Flow Offload is supported only on native instances of the ASA and FTD applications and on FPR4100 and FPR9300 platforms. FTD container instances do not support flow offload.

---

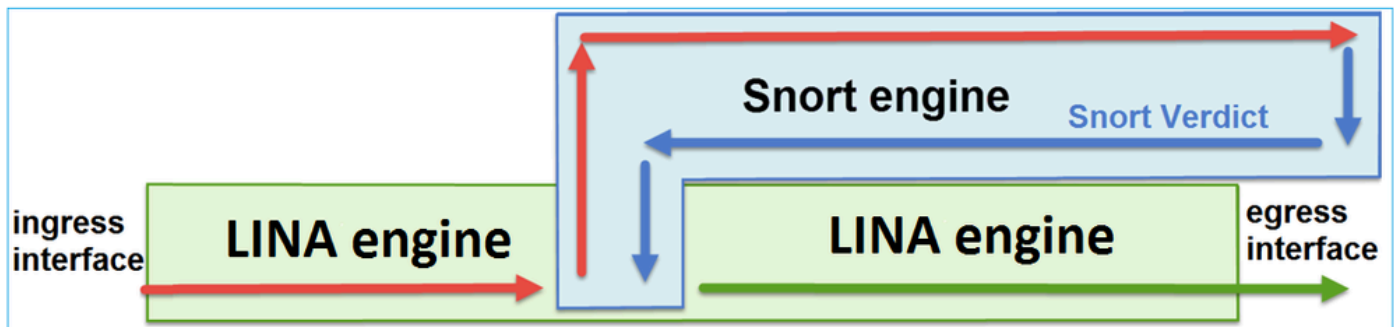
## Background Information

The background operation of each action is examined along with its interaction with other features like Flow Offload and protocols that open secondary connections.

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

- LINA engine
- 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 (permit list or block list) for the packet
- The LINA engine drops or forwards the packet based on Snort's verdict

## How ACP is Deployed

The FTD policy is configured on FMC when off-box (remote) management is used or Firepower Device Manager (FDM) when local management is used. In both scenarios, the ACP is deployed as:

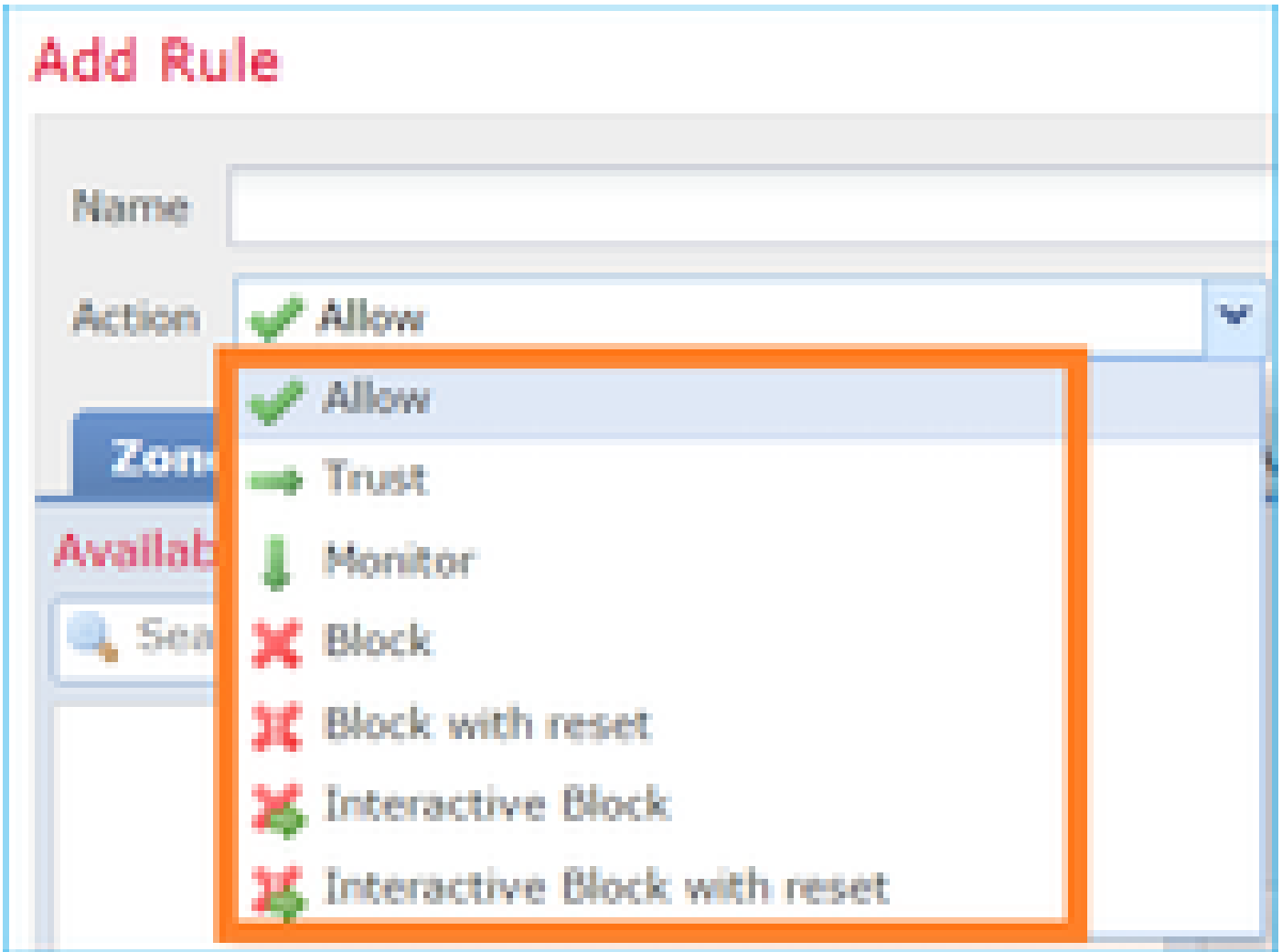
- A global Access Control List (ACL) named CSM\_FW\_ACL\_ to the FTD LINA engine
- Access Control (AC) rules in the /ngfw/var/sf/detection\_engines/<UUID>/ngfw.rules file to the FTD Snort engine

## Configure

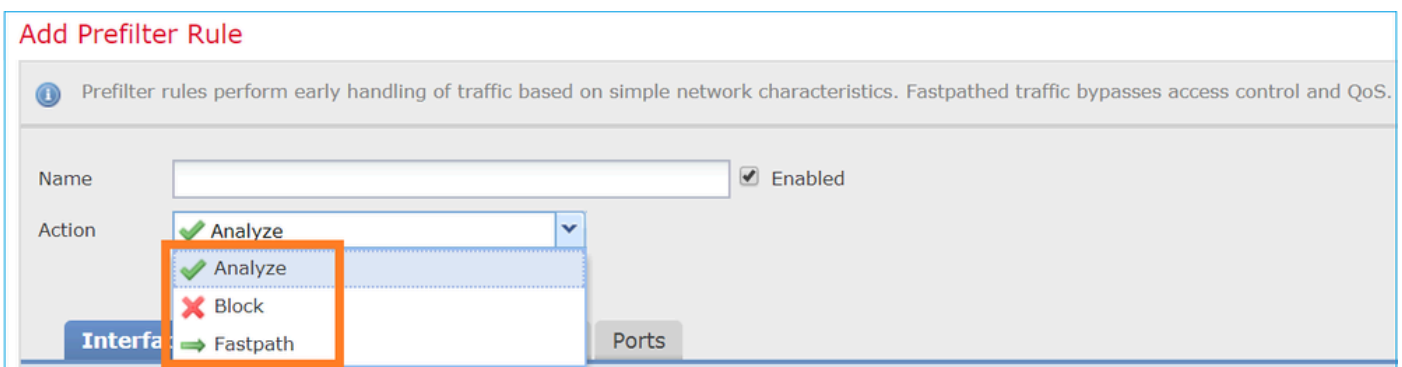
### ACP Available Actions

The FTD ACP contains one or more rules and each rule can have one of these actions and as shown in the image:

- **Allow**
- **Trust**
- **Monitor**
- **Block**
- **Block with reset**
- **Interactive Block**
- **Interactive Block with reset**



Similarly, a Prefilter Policy can contain one or more rules and the possible actions are shown in the image:



## How ACP and Prefilter Policy Interact

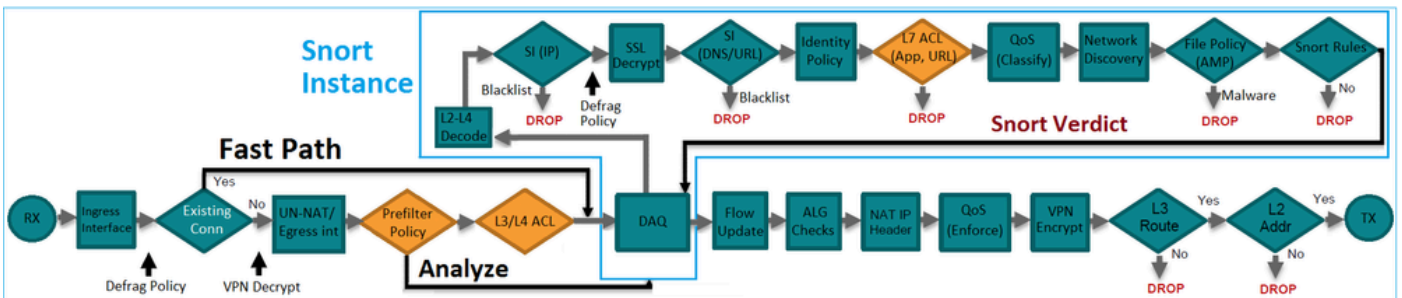
The Prefilter Policy was introduced in the 6.1 version and serves 2 main purposes:

1. It allows the inspection of tunneled traffic where the FTD LINA engine checks the outer IP header while the Snort engine checks the inner IP header. More specifically, in the case of tunneled traffic (for example GRE) the rules in the Prefilter Policy always act on the outer headers, while the rules in the ACP are always applicable to the inside sessions (inner headers). The tunneled traffic refers to these protocols:

- GRE

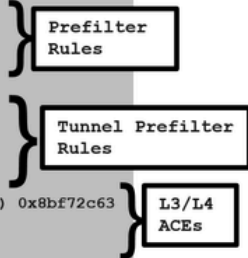
- IP-in-IP
- IPv6-in-IP
- Teredo Port 3544

2. It provides Early Access Control (EAC) which allows the flow to completely bypass the Snort engine as shown in the image.



The Prefilter Rules are deployed on FTD as L3/L4 Access Control Elements (ACEs) and precede the configured L3/L4 ACEs as shown in the image:

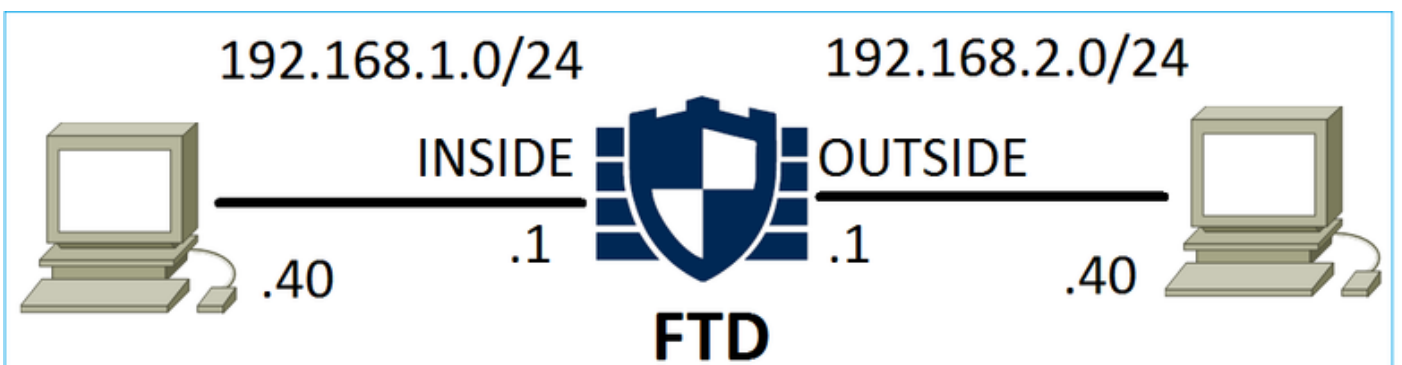
```
firepower# show access-list
access-list CSM_FW_ACL_line 1 remark rule-id 268434457: PREFILTER POLICY: FTD_Prefilter_Policy
access-list CSM_FW_ACL_line 2 remark rule-id 268434457: RULE: Fastpath_Rule1
access-list CSM_FW_ACL_line 3 advanced trust ip host 192.168.75.16 any rule-id 268434457 event-log both (hitcnt=0)
access-list CSM_FW_ACL_line 4 remark rule-id 268434456: PREFILTER POLICY: FTD_Prefilter_Policy
access-list CSM_FW_ACL_line 5 remark rule-id 268434456: RULE: DEFAULT TUNNEL ACTION RULE
access-list CSM_FW_ACL_line 6 advanced permit ipinip any any rule-id 268434456 (hitcnt=0) 0xf5b597d6
access-list CSM_FW_ACL_line 7 advanced permit 41 any any rule-id 268434456 (hitcnt=0) 0x06095aba
access-list CSM_FW_ACL_line 8 advanced permit gre any any rule-id 268434456 (hitcnt=2) 0x52c7a066
access-list CSM_FW_ACL_line 9 advanced permit udp any any eq 3544 rule-id 268434456 (hitcnt=0) 0xcf6309bc
access-list CSM_FW_ACL_line 10 remark rule-id 268434445: ACCESS POLICY: FTD5506-1 - Mandatory/1
access-list CSM_FW_ACL_line 12 advanced deny ip host 10.1.1.1 any rule-id 268434445 event-log flow-start (hitcnt=0) 0x8bf72c63
access-list CSM_FW_ACL_line 14 remark rule-id 268434434: L4 RULE: DEFAULT ACTION RULE
access-list CSM_FW_ACL_line 15 advanced permit ip any any rule-id 268434434 (hitcnt=410) 0xa1d3780e
```



**Note:** Prefilter v/s ACP rules = the first match is applied.

### ACP Block Action

Consider the topology shown in this image:



### Scenario 1. Early LINA Drop

The ACP contains a Block rule which uses an L4 condition (Destination Port TCP 80) as shown in the image:

Access Control													
ACP1													
Enter Description													
Prefilter Policy: <a href="#">Default Prefilter Policy</a>				SSL Policy: <a href="#">None</a>				Identity Policy: <a href="#">None</a>					
Rules													
Filter by Device													
#	Name	Source Zones	Dest Zones	Source Networks	Dest Networks	VLAN	Users	Applica...	Source ...	Dest Ports	URLs	ISE/SGT Attribu...	Action
Mandatory - ACP1 (1-1)													
1	Rule1	Any	Any	192.168.1.40	192.168.2.40	Any	Any	Any	Any	TCP (6):80	Any	Any	Block

The deployed policy in Snort:

```
<#root>
```

```
268435461
```

```
deny
```

```
any 192.168.1.40 32 any any 192.168.2.40 32 80 any 6
```

The deployed policy in LINA. Note that the rule is pushed as deny action:

```
<#root>
```

```
firepower#
```

```
show access-list
```

```
...
access-list CSM_FW_ACL_ line 9 remark rule-id 268435461: L4 RULE: Rule1
access-list CSM_FW_ACL_ line 10 advanced
```

```
deny
```

```
tcp host 192.168.1.40 host 192.168.2.40 eq www rule-id 268435461 event-log flow-start (hitcnt=0) 0x614
```

### Verify Behavior:

When host-A (192.168.1.40) tries to open an HTTP session to host-B (192.168.2.40) the TCP synchronize (SYN) packets are dropped by the FTD LINA engine and do not reach the Snort Engine or the destination:

```
<#root>
```

```
firepower#
```

```
show capture
```

```
capture CAPI type raw-data buffer 33554432 trace trace-count 100 interface
```

```
INSIDE
```

```
[Capturing -
```

```
430 bytes
```

```
]
  match ip host 192.168.1.40 any
capture CAPO type raw-data buffer 33554432 trace trace-count 100 interface
OUTSIDE

[Capturing -
0 bytes
]
```

<#root>

firepower#

show capture CAPI

```
1: 11:08:09.672801 192.168.1.40.32789 > 192.168.2.40.80: S 3249160620:3249160620(0) win 2920 <mss 1
2: 11:08:12.672435 192.168.1.40.32789 > 192.168.2.40.80: S 3249160620:3249160620(0) win 2920 <mss 1
3: 11:08:18.672847 192.168.1.40.32789 > 192.168.2.40.80: S 3249160620:3249160620(0) win 2920 <mss 1
4: 11:08:30.673610 192.168.1.40.32789 > 192.168.2.40.80: S 3249160620:3249160620(0) win 2920 <mss 1
```

<#root>

firepower#

show capture CAPI packet-number 1 trace

```
1: 11:08:09.672801 192.168.1.40.32789 > 192.168.2.40.80:
```

```
s
3249160620:3249160620(0) win 2920 <mss 1460,sackOK,timestamp 4060517 0>
...
```

Phase: 4

Type: ACCESS-LIST

Subtype: log

Result: DROP

Config:

access-group CSM\_FW\_ACL\_ global

access-list CSM\_FW\_ACL\_ advanced deny tcp host 192.168.1.40 host 192.168.2.40 eq www rule-id 268435461 e

access-list CSM\_FW\_ACL\_ remark rule-id 268435461: ACCESS POLICY: ACP1 - Mandatory

access-list CSM\_FW\_ACL\_ remark rule-id 268435461: L4 RULE: Rule1

Additional Information:

<- No Additional Information = No Snort Inspection

Result:

input-interface: INSIDE

input-status: up

input-line-status: up

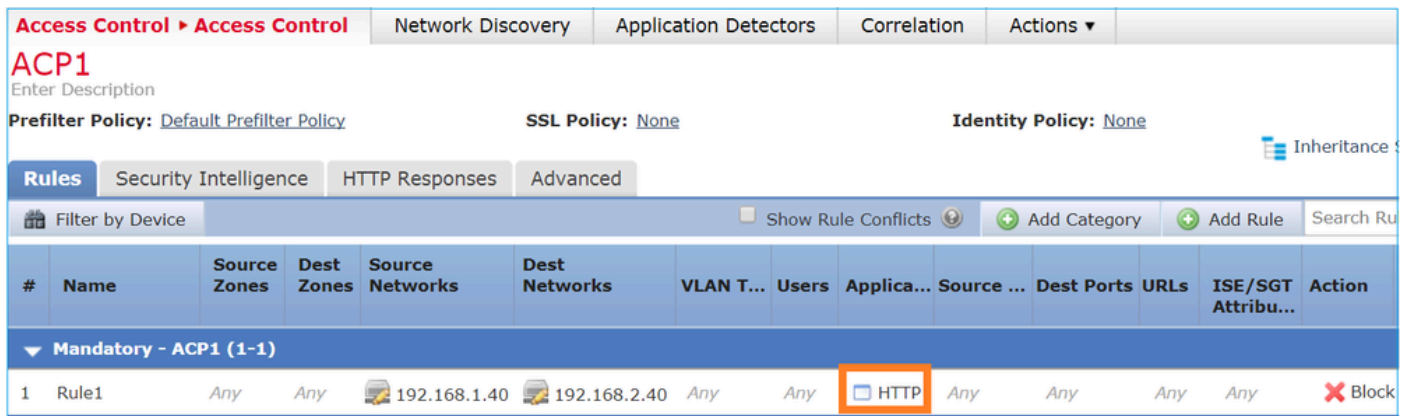
output-interface: OUTSIDE

output-status: up  
output-line-status: up  
Action: drop

Drop-reason: (acl-drop) Flow is denied by configured rule

## Scenario 2. Drop Due to Snort Verdict

The ACP contains a Block rule which uses an L7 condition (Application HTTP) as shown in the image:



#	Name	Source Zones	Dest Zones	Source Networks	Dest Networks	VLAN T...	Users	Applica...	Source ...	Dest Ports	URLs	ISE/SGT Attribu...	Action
1	Rule1	Any	Any	192.168.1.40	192.168.2.40	Any	Any	HTTP	Any	Any	Any	Any	Block


The deployed policy in Snort:

```
<#root>  
268435461  
deny  
  any 192.168.1.40 32 any any 192.168.2.40 32 any any any  
(appid 676:1)
```

Appid 676:1 = HTTP

The deployed policy in LINA.

---

 **Note:** The rule is pushed as a **permit** action because LINA cannot determine that the session uses HTTP. On FTD the Application Detection mechanism is in the Snort engine.

---


```
<#root>  
firepower#  
show access-list  
...  
access-list CSM_FW_ACL_ line 9 remark rule-id 268435461: L7 RULE: Rule1  
access-list CSM_FW_ACL_ line 10 advanced  
  
permit
```



```
ip host 192.168.1.40 host 192.168.2.40 rule-id 268435461 (hitcnt=0) 0xb788b786
```

For a Block rule that uses **Application** as a condition, the trace of a real packet shows that the session is dropped by the LINA due to the Snort engine verdict.

---

 **Note:** In order for the Snort engine to determine the application it has to inspect a few packets (usually 3-10 which depends on the application decoder). Thus a few packets are allowed through the FTD and they make it to the destination. The allowed packets are still subject to the Intrusion Policy check based on the **Access Policy > Advanced > Intrusion Policy used before Access Control rule is determined** option.

---

### Verify Behavior:

When host-A (192.168.1.40) tries to establish an HTTP session with host-B (192.168.2.40) the LINA ingress capture shows:

```
<#root>
firepower#
show capture CAPI

8 packets captured

  1: 11:31:19.825564 192.168.1.40.32790 > 192.168.2.40.80:
s
357753151:357753151(0) win 2920 <mss 1460,sackOK,timestamp 5450579 0>
  2: 11:31:19.826403 192.168.2.40.80 > 192.168.1.40.32790:
s
1283931030:1283931030(0)
ack
357753152 win 2896 <mss 1380,sackOK,timestamp 5449236 5450579>
  3: 11:31:19.826556 192.168.1.40.32790 > 192.168.2.40.80: P 357753152:357753351(199)
ack
1283931031 win 2920 <nop,nop,timestamp 5450580 5449236>
  4: 11:31:20.026899 192.168.1.40.32790 > 192.168.2.40.80: P 357753152:357753351(199) ack 1283931031
  5: 11:31:20.428887 192.168.1.40.32790 > 192.168.2.40.80: P 357753152:357753351(199) ack 1283931031
...
```

The egress capture:

```
<#root>
firepower#
show capture CAPO
```

5 packets captured

```
1: 11:31:19.825869 192.168.1.40.32790 > 192.168.2.40.80: S 1163713179:1163713179(0) win 2920 <mss 1
2: 11:31:19.826312 192.168.2.40.80 > 192.168.1.40.32790: S 354801457:354801457(0) ack 1163713180 wi
3: 11:31:23.426049 192.168.2.40.80 > 192.168.1.40.32790: S 354801457:354801457(0) ack 1163713180 wi
4: 11:31:29.426430 192.168.2.40.80 > 192.168.1.40.32790: S 354801457:354801457(0) ack 1163713180 wi
5: 11:31:41.427208 192.168.2.40.80 > 192.168.1.40.32790: S 354801457:354801457(0) ack 1163713180 wi
```

The trace shows that the first packet (TCP SYN) is allowed by the Snort since the Application Detection verdict has not been reached yet:

<#root>

firepower#

show capture CAPI packet-number 1 trace

```
1: 11:31:19.825564 192.168.1.40.32790 > 192.168.2.40.80:
```

s

```
357753151:357753151(0) win 2920 <mss 1460,sackOK,timestamp 5450579 0>
```

...

Phase: 4

Type: ACCESS-LIST

Subtype: log

Result: ALLOW

Config:

```
access-group CSM_FW_ACL_ global
```

```
access-list CSM_FW_ACL_ advanced permit ip host 192.168.1.40 host 192.168.2.40 rule-id 268435461
```

```
access-list CSM_FW_ACL_ remark rule-id 268435461: ACCESS POLICY: ACP1 - Mandatory
```

```
access-list CSM_FW_ACL_ remark rule-id 268435461: L7 RULE: Rule1
```

Additional Information:

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

...

Phase: 10

Type: FLOW-CREATION

Subtype:

Result: ALLOW

Config:

Additional Information:

New flow created with id 23194

, packet dispatched to next module

...

Phase: 12

Type: SNORT

Subtype:

Result: ALLOW

Config:

Additional Information:

Snort Trace:

```
Packet: TCP, SYN, seq 357753151
```

```
AppID: service unknown (0), application unknown (0)
```

```
Firewall: starting rule matching, zone -1 -> -1, geo 0 -> 0, vlan 0, sgt 65535, user 9999997, icmpType 0
```

Firewall:

pending rule-matching, id 268435461, pending AppID

NAP id 1, IPS id 0,

Verdict PASS

Snort Verdict: (pass-packet) allow this packet

Result:

input-interface: OUTSIDE

input-status: up

input-line-status: up

output-interface: OUTSIDE

output-status: up

output-line-status: up

Action: allow

The same for the TCP SYN/ACK packet:

<#root>

firepower#

show capture CAPO packet-number 2 trace

2: 11:31:19.826312 192.168.2.40.80 > 192.168.1.40.32790:

s

354801457:354801457(0)

ack

1163713180 win 2896 <mss 1460,sackOK,timestamp 5449236 5450579>

...

Phase: 3

Type: FLOW-LOOKUP

Subtype:

Result: ALLOW

Config:

Additional Information:

Found flow with id 23194, using existing flow

...

Phase: 5

Type: SNORT

Subtype:

Result: ALLOW

Config:

Additional Information:

Snort Trace:

Packet: TCP, SYN, ACK, seq 1283931030, ack 357753152

AppID: service unknown (0), application unknown (0)

Firewall: starting rule matching, zone -1 -> -1, geo 0 -> 0, vlan 0, sgt 65535, user 9999997, icmpType 0

Firewall: pending rule-matching, id 268435461, pending AppID

NAP id 1, IPS id 0,

Verdict PASS

Snort Verdict: (pass-packet) allow this packet

Result:

input-interface: INSIDE

input-status: up

input-line-status: up

output-interface: INSIDE

output-status: up

output-line-status: up

Action: allow

Snort returns a DROP verdict once an inspection of the third packet completes:

<#root>

firepower#

show capture CAPI packet-number 3 trace

3: 11:31:19.826556 192.168.1.40.32790 > 192.168.2.40.80: P 357753152:357753351(199)

ack

1283931031 win 2920 <nop,nop,timestamp 5450580 5449236>

Phase: 3

Type: FLOW-LOOKUP

Subtype:

Result: ALLOW

Config:

Additional Information:

Found flow with id 23194, using existing flow

Phase: 5

Type: SNORT

Subtype:

Result: DROP

Config:

Additional Information:

Snort Trace:

Packet: TCP, ACK, seq 357753152, ack 1283931031

AppID: service HTTP (676)

, application unknown (0)

Firewall: starting rule matching, zone -1 -> -1, geo 0(0) -> 0, vlan 0, sgt 65535, user 9999997, url ht

Firewall: block rule, id 268435461, drop

Snort: processed decoder alerts or actions queue, drop

NAP id 1, IPS id 0,

Verdict BLOCKLIST, Blocked by Firewall

Snort Verdict: (block-list) block list this flow

Result:

input-interface: INSIDE

input-status: up

input-line-status: up

Action: drop

Drop-reason: (firewall) Blocked by the firewall preprocessor

You can also run the command `system support trace` from FTD CLISH mode. This tool provides 2 functions:

- Shows the Snort verdict for each packet as it is sent to the Data Acquisition library (DAQ) and seen in LINA. DAQ is a component located between the FTD LINA engine and the Snort engine
- Allows to run `system support firewall-engine-debug` at the same time to see what happens within the Snort engine itself

Here is the output:

```
<#root>
```

```
>
```

```
system support trace
```

```
Please specify an IP protocol:
```

```
tcp
```

```
Please specify a client IP address:
```

```
192.168.1.40
```

```
Please specify a client port:
```

```
Please specify a server IP address:
```

```
192.168.2.40
```

```
Please specify a server port:
```

```
Enable firewall-engine-debug too? [n]:
```

```
y
```

```
Monitoring packet tracer debug messages
```

```
Tracing enabled by Lina
```

```
192.168.2.40-80 - 192.168.1.40-32791 6 Packet:
```

```
TCP, SYN
```

```
, seq 2620409313
```

```
192.168.2.40-80 - 192.168.1.40-32791 6 AppID: service unknown (0), application unknown (0)
```

```
192.168.1.40-32791 > 192.168.2.40-80 6 AS 1 I 0 New session
```

```
192.168.1.40-32791 > 192.168.2.40-80 6 AS 1 I 0 Starting with minimum 2, 'Rule1', and SrcZone first with
```

```
192.168.1.40-32791 > 192.168.2.40-80 6 Firewall: starting rule matching, zone -1 -> -1, geo 0 -> 0, vla
```

```
192.168.1.40-32791 > 192.168.2.40-80 6 AS 1 I 0 pending rule order 2, 'Rule1', AppID
```

```
192.168.1.40-32791 > 192.168.2.40-80 6 Firewall:
```

```
pending rule-matching, 'Rule1', pending AppID
```

192.168.1.40-32791 > 192.168.2.40-80 6 NAP id 1, IPS id 0,

Verdict PASS

Trace buffer and verdict reason are sent to DAQ's PDTS

Tracing enabled by Lina

192.168.2.40-80 - 192.168.1.40-32791 6 Packet:

TCP, SYN, ACK

, seq 3700371680, ack 2620409314

192.168.2.40-80 - 192.168.1.40-32791 6 AppID: service unknown (0), application unknown (0)

192.168.1.40-32791 > 192.168.2.40-80 6 AS 1 I 0 Starting with minimum 2, 'Rule1', and SrcZone first with

192.168.1.40-32791 > 192.168.2.40-80 6 Firewall: starting rule matching, zone -1 -> -1, geo 0 -> 0, vla

192.168.1.40-32791 > 192.168.2.40-80 6 AS 1 I 0 pending rule order 2, 'Rule1', AppID

192.168.1.40-32791 > 192.168.2.40-80 6 Firewall:

pending rule-matching, 'Rule1', pending AppID

192.168.1.40-32791 > 192.168.2.40-80 6 NAP id 1, IPS id 0,

Verdict PASS

Trace buffer and verdict reason are sent to DAQ's PDTS

Tracing enabled by Lina

192.168.2.40-80 - 192.168.1.40-32791 6 Packet:

TCP, ACK

, seq 2620409314, ack 3700371681

192.168.2.40-80 - 192.168.1.40-32791 6 AppID:

service HTTP (676)

, application unknown (0)

192.168.1.40-32791 > 192.168.2.40-80 6 AS 1 I 0 Starting with minimum 2, 'Rule1', and SrcZone first with

192.168.1.40-32791 > 192.168.2.40-80 6 Firewall: starting rule matching, zone -1 -> -1, geo 0(0) -> 0, v

url http://192.168.2.40/128k.html

192.168.1.40-32791 > 192.168.2.40-80 6 AS 1 I 0

match rule order 2, 'Rule1', action Block

192.168.1.40-32791 > 192.168.2.40-80 6 AS 1 I 0 deny action

192.168.1.40-32791 > 192.168.2.40-80 6

Firewall: block rule, 'Rule1', drop

192.168.1.40-32791 > 192.168.2.40-80 6 Snort: processed decoder alerts or actions queue, drop

192.168.1.40-32791 > 192.168.2.40-80 6 AS 1 I 0 Deleting session

192.168.1.40-32791 > 192.168.2.40-80 6 NAP id 1, IPS id 0,

Verdict BLOCKLIST

192.168.1.40-32791 > 192.168.2.40-80 6 ==>

Blocked by Firewall

## Summary

- The ACP Block Action gets deployed as either permit or deny rule in LINA which depends on the rule conditions

- If the conditions are L3/L4 then the LINA blocks the packet. In the case of TCP, the first packet (TCP SYN) is blocked
- If the conditions are L7 then the packet is forwarded to the Snort engine for further inspection. In the case of TCP, a few packets are allowed through FTD until Snort reaches a verdict. The allowed packets are still subject to the Intrusion Policy check based on the **Access Policy > Advanced > Intrusion Policy used before Access Control rule is determined** option.

## ACP Block with reset Action

A Block with rest rule configured on FMC UI:

Name	Source Zones	Dest Zones	Source Networks	Dest Networks	VLAN Tags	Users	Applic...	Source Ports	Dest Ports	URLs	Source SGT	Dest SGT	Action
Mandatory - ACP1 (1-4)													
1	Block-RST-Rule1	Any	Any	192.168.10.0/24	192.168.11.50	Any	Any	Any	TCP (6):80	Any	Any	Any	Block with reset
2	Block-RST_Rule2	Any	Any	192.168.10.0/24	192.168.11.51	Any	Any	HTTP	Any	Any	Any	Any	Block with reset

The Block with reset rule is deployed on FTD LINA engine as a **permit** and to Snort engine as a **reset** rule:

```
<#root>
```

```
firepower#
```

```
show access-list
```

```
...
```

```
access-list CSM_FW_ACL_ line 10 advanced
```

```
permit
```

```
tcp 192.168.10.0 255.255.255.0 host 192.168.11.50 eq www rule-id 268438864 (hitcnt=0) 0xba785fc0
access-list CSM_FW_ACL_ line 11 remark rule-id 268438865: ACCESS POLICY: ACP1 - Mandatory
access-list CSM_FW_ACL_ line 12 remark rule-id 268438865: L7 RULE: Block-RST_Rule2
access-list CSM_FW_ACL_ line 13 advanced
```

```
permit
```

```
ip 192.168.10.0 255.255.255.0 host 192.168.11.51 rule-id 268438865 (hitcnt=0) 0x622350d0
```

Snort engine:

```
<#root>
```

```
admin@firepower:~$
```

```
cat /var/sf/detection_engines/9e080e5c-adc3-11ea-9d37-44884cf7e9ba/ngfw.rules
```

```
...
```

```
# Start of AC rule.
```

```
268438864
```

```
reset
```

```
any 192.168.10.0 24 any any 192.168.11.50 32 80 any 6
# End rule 268438864
268438865
```

reset

```
any 192.168.10.0 24 any any 192.168.11.51 32 any any any (appid 676:1) (ip_protos 6, 17)
# End rule 268438865
```

When a packet matches Block with reset rule FTD sends a TCP Reset packet or an ICMP Type 3 Code 13 Destination Unreachable (Administratively filtered) message:

<#root>

```
root@kali:~/tests#
```

```
wget 192.168.11.50/file1.zip
```

```
--2020-06-20 22:48:10-- http://192.168.11.50/file1.zip
Connecting to 192.168.11.50:80... failed:
```

```
Connection refused.
```

Here is a capture taken on the FTD ingress interface:

```
firepower# show capture CAPI
```

```
2 packets captured
```

```
1: 21:01:00.977259 802.1Q vlan#202 PO 192.168.10.50.41986 > 192.168.11.50.80: S 3120295488:3120295488(0
```

```
2: 21:01:00.978114 802.1Q vlan#202 PO 192.168.11.50.80 > 192.168.10.50.41986: R 0:0(0) ack 3120295489 w
```

System support trace output, in this case, shows that the packet is dropped due to the Snort verdict:

<#root>

>

```
system support trace
```

```
Enable firewall-engine-debug too? [n]:
```

```
y
```

```
Please specify an IP protocol: tcp
```

```
Please specify a client IP address:
```

```
192.168.10.50
```

```
Please specify a client port:
```

```
Please specify a server IP address:
```

```
192.168.11.50
```

```
Please specify a server port:
```

```
Monitoring packet tracer and firewall debug messages
```

```
192.168.10.50-41984 - 192.168.11.50-80 6 AS 1-1 CID 0 Packet: TCP, SYN, seq 3387496622
```

```
192.168.10.50-41984 - 192.168.11.50-80 6 AS 1-1 CID 0
```



Session: new snort session

```
192.168.10.50-41984 - 192.168.11.50-80 6 AS 1-1 CID 0 AppID: service unknown (0), application unknown (0)
192.168.10.50-41984 > 192.168.11.50-80 6 AS 1-1 I 9
```

new firewall session

```
192.168.10.50-41984 > 192.168.11.50-80 6 AS 1-1 I 9 using HW or preset rule order 2, 'Block-RST-Rule1',
192.168.10.50-41984 > 192.168.11.50-80 6 AS 1-1 I 9 HitCount data sent for rule id: 268438864,
192.168.10.50-41984 > 192.168.11.50-80 6 AS 1-1 I 9
```

reset action

```
192.168.10.50-41984 > 192.168.11.50-80 6 AS 1-1 I 9 deleting firewall session flags = 0x0, fwFlags = 0x0
192.168.10.50-41984 - 192.168.11.50-80 6 AS 1-1 CID 0
```

Firewall: block w/ reset rule, 'Block-RST-Rule1', drop

```
192.168.10.50-41984 - 192.168.11.50-80 6 AS 1-1 CID 0 Snort: processed decoder alerts or actions queue,
192.168.10.50-41984 - 192.168.11.50-80 6 AS 1-1 CID 0 Snort id 9, NAP id 1, IPS id 0,
```

Verdict BLOCKLIST

```
192.168.10.50-41984 - 192.168.11.50-80 6 AS 1-1 CID 0 ==>
```

Blocked by Firewall

Verdict reason is sent to DAQ

## Use Cases

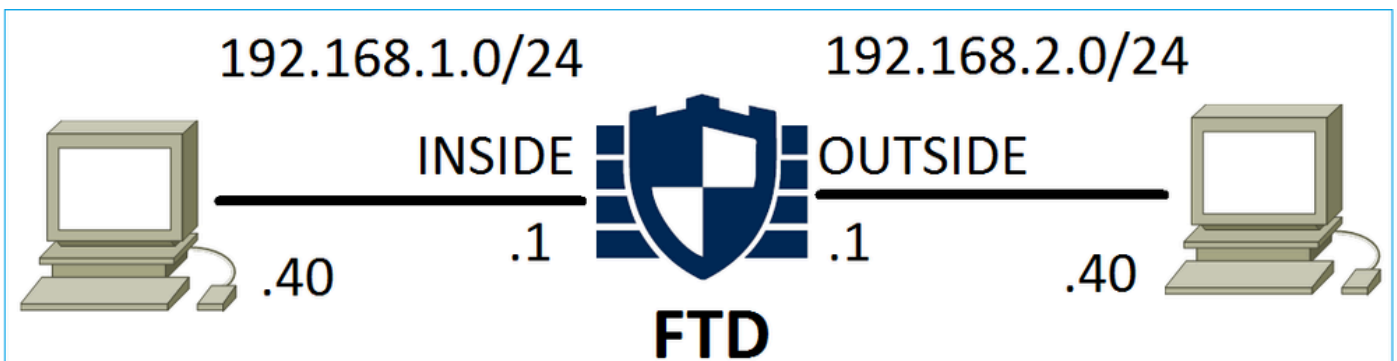
Same as **Block** action, but terminates immediately the connection.

## ACP Allow Action

### Scenario 1. ACP Allow Action (L3/L4 Conditions)

Normally, you would configure an **Allow** rule to specify additional inspections like an Intrusion Policy and/or a File Policy. This first scenario demonstrates the operation of an Allow rule when an L3/L4 condition is applied.

Consider this topology as shown in the image:



This policy is applied as shown in the image:


Access Control > Access Control													
Network Discovery			Application Detectors			Correlation			Actions				
<b>ACP1</b>													
Enter Description													
Prefilter Policy: <a href="#">Default Prefilter Policy</a>				SSL Policy: <a href="#">None</a>				Identity Policy: <a href="#">None</a>					
<a href="#">Inheritance Settings</a>													
<b>Rules</b>   Security Intelligence   HTTP Responses   Advanced													
Filter by Device <input type="checkbox"/> Show Rule Conflicts <input type="checkbox"/> Add Category <input type="checkbox"/> Add Rule Search Rules													
#	Name	Source Zones	Dest Zones	Source Networks	Dest Networks	VLAN T...	Users	Applica...	Source ...	Dest Ports	URLs	ISE/SGT Attribu...	Action
Mandatory - ACP1 (1-1)													
1	Rule1	Any	Any	192.168.1.40	192.168.2.40	Any	Any	Any	Any	TCP (6):80	Any	Any	Allow

The deployed policy in Snort. Note that the rule is deployed as an **allow** action:

```
<#root>
# Start of AC rule.
268435461
allow
any 192.168.1.40 32 any any 192.168.2.40 32 80 any 6
```

The policy in LINA.

---

 **Note:** The rule is deployed as a **permit** action which essentially means redirection to Snort for further inspection.

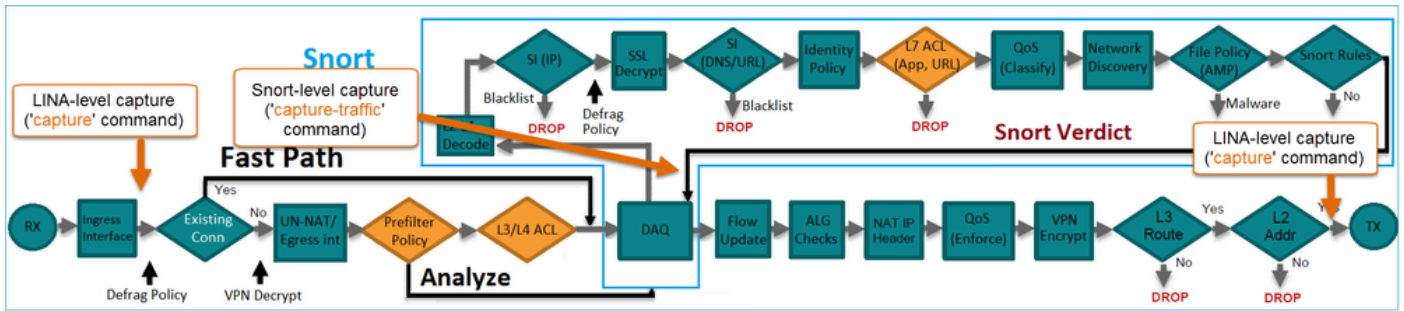
---

```
<#root>
firepower#
show access-list
...
access-list CSM_FW_ACL_ line 9 remark rule-id 268435461: L7 RULE: Rule1
access-list CSM_FW_ACL_ line 10 advanced
permit
tcp host 192.168.1.40 host 192.168.2.40 eq www rule-id 268435461 (hitcnt=1) 0x641a20c3
```

In order to see how FTD handles a flow that matches an **Allow** rule there are a few ways:

- Verify Snort Statistics
- With the use of system support trace CLISH tool
- With the use of capture with the trace option in LINA and optionally with capture-traffic in Snort engine

LINA capture vs Snort capture-traffic:



## Verify Behavior:

Clear the Snort statistics, enable **system support trace** from CLISH, and initiate an HTTP flow from host-A (192.168.1.40) to host-B (192.168.2.40). All the packets are forwarded to the Snort engine and get the PASS verdict by the Snort:

```
<#root>
```

```
firepower#
```

```
clear snort statistics
```

```
<#root>
```

```
>
```

```
system support trace
```

```
Please specify an IP protocol:
```

```
Please specify a client IP address:
```

```
192.168.1.40
```

```
Please specify a client port:
```

```
Please specify a server IP address:
```

```
192.168.2.40
```

```
Please specify a server port:
```

```
Enable firewall-engine-debug too? [n]:
```

```
Monitoring packet tracer debug messages
```

```
Tracing enabled by Lina
```

```
192.168.2.40-80 - 192.168.1.40-32797 6 Packet: TCP, SYN, seq 361134402
```

```
192.168.2.40-80 - 192.168.1.40-32797 6 AppID: service unknown (0), application unknown (0)
```

```
192.168.1.40-32797 > 192.168.2.40-80 6 Firewall: allow rule, 'Rule1', allow
```

```
192.168.1.40-32797 > 192.168.2.40-80 6 NAP id 1, IPS id 0,
```

```
Verdict PASS
```

```
Trace buffer and verdict reason are sent to DAQ's PDTS
```

```
Tracing enabled by Lina
```

```
192.168.2.40-80 - 192.168.1.40-32797 6 Packet: TCP, SYN, ACK, seq 1591434735, ack 361134403
```

```
192.168.2.40-80 - 192.168.1.40-32797 6 AppID: service unknown (0), application unknown (0)
```

```
192.168.1.40-32797 > 192.168.2.40-80 6 Firewall: allow rule, 'Rule1', allow
```

```
192.168.1.40-32797 > 192.168.2.40-80 6 NAP id 1, IPS id 0,
```

```
Verdict PASS
```

```
Trace buffer and verdict reason are sent to DAQ's PDTS
```

Tracing enabled by Lina

```
192.168.2.40-80 - 192.168.1.40-32797 6 Packet: TCP, ACK, seq 361134403, ack 1591434736
192.168.2.40-80 - 192.168.1.40-32797 6 AppID: service HTTP (676), application unknown (0)
192.168.1.40-32797 > 192.168.2.40-80 6 Firewall: allow rule, 'Rule1', allow
192.168.1.40-32797 > 192.168.2.40-80 6 NAP id 1, IPS id 0,
```

Verdict PASS

The Pass Packets counters increase:

```
<#root>
```

```
>
```

```
show snort statistics
```

Packet Counters:

Passed Packets	54
Blocked Packets	0
Injected Packets	0
Packets bypassed (Snort Down)	0
Packets bypassed (Snort Busy)	0

Flow Counters:

Fast-Forwarded Flows	0
Blocklisted Flows	0

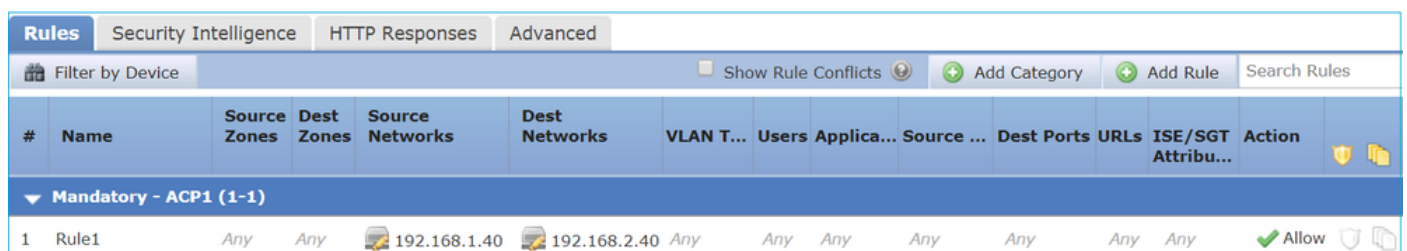
...

Passed Packets = Inspected by the Snort engine

## Scenario 2. ACP Allow Action (L3-7 Conditions)

Similar behavior is seen when the Allow rule is deployed as follows.

Only an L3/L4 condition as shown in the image:



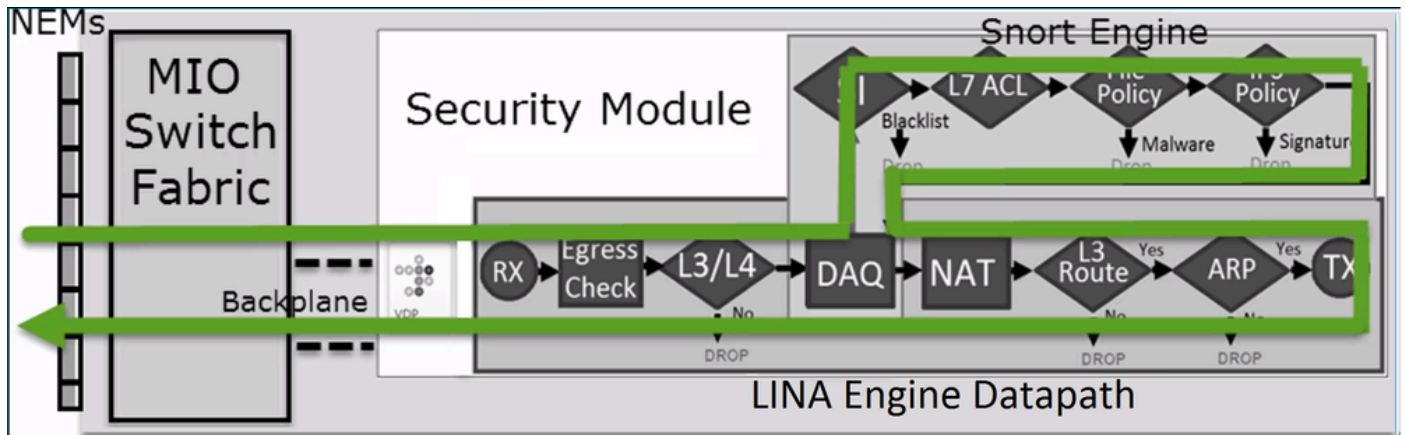
#	Name	Source Zones	Dest Zones	Source Networks	Dest Networks	VLAN T...	Users	Applica...	Source ...	Dest Ports	URLs	ISE/SGT Attribu...	Action
Mandatory - ACP1 (1-1)													
1	Rule1	Any	Any	192.168.1.40	192.168.2.40	Any	Any	Any	Any	Any	Any	Any	Allow

An L7 condition (for example Intrusion Policy, File Policy, Application, etc) is shown in the image:

#	Name	Source Zones	Dest Zones	Source Networks	Dest Networks	VLAN T...	Users	Applica...	Source ...	Dest Ports	URLs	ISE/SGT Attribu...	Action
Mandatory - ACP1 (1-1)													
1	Rule1	Any	Any	192.168.1.40	192.168.2.40	Any	Any	Any	Any	Any	Any	Any	Allow

## Summary

In order to summarize, this is how a flow is handled by an FTD deployed on an FP4100/9300 when an **Allow** rule is matched as shown in the image:



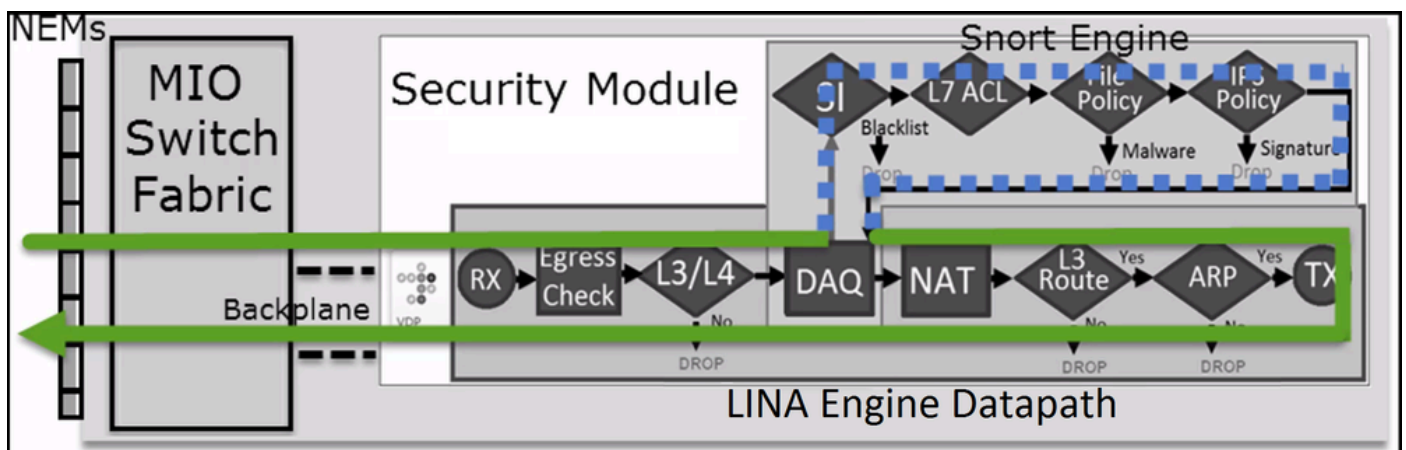
**Note:** Management Input Output (MIO) is the Supervisor engine of the firepower chassis.

## Scenario 3. Snort Fast-Forward verdict with Allow

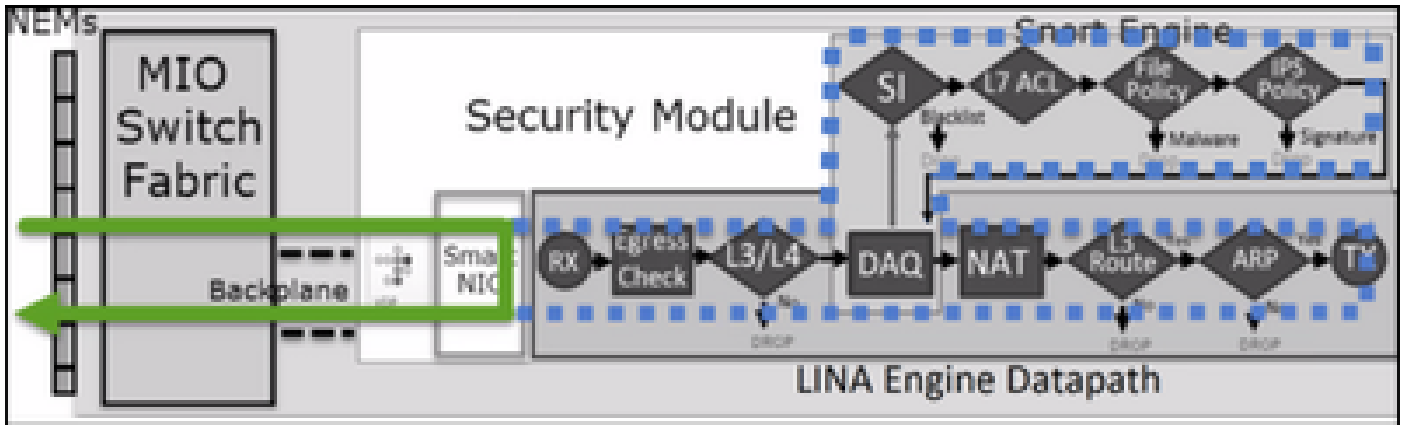
There are specific scenarios where the FTD Snort engine gives a PERMITLIST verdict (fast-forward) and the rest of the flow is offloaded to the LINA engine (in some cases then is offloaded to the HW Accelerator - SmartNIC). These are:

1. SSL traffic without an SSL policy configured
2. Intelligent application bypass (IAB)

This is the visual representation of the packet path:



Or in some cases:



## Main Points

- The Allow Rule is deployed as **allow** in Snort and **permit** in LINA
- In most cases, all the packets of a session are forwarded to the Snort engine for additional inspection

## Use Cases

You would configure an **Allow** rule when you need L7 inspection by Snort Engine such as:

- Intrusion Policy
- File Policy

## ACP Trust Action

### Scenario 1. ACP Trust Action

If you do not want to apply advanced L7 inspection at the Snort level (for example Intrusion Policy, File Policy, Network Discovery), but you still want to use features like Security Intelligence (SI), Identity Policy, QoS, etc, then it is recommended to use the Trust action in your rule.

Topology:




The configured policy:

Name	Source Zones	Dest Zones	Source Networks	Dest Networks	VLAN Tags	Users	Applic...	Source Ports	Dest Ports	URLs	Source SGT	Dest SGT	Action	Icons
Mandatory - ACP1 (1-4)														
1	trust_L3-L4	Any	Any	192.168.10.50 192.168.10.51	192.168.11.50 192.168.11.51	Any	Any	Any	Any	TCP (6):80	Any	Any	Trust	Icons

The Trust rule as it is deployed in FTD Snort engine:

```
<#root>
# Start of AC rule.
268438858
fastpath
  any 192.168.10.50 31 any any 192.168.11.50 31
80
any
6
(log dcforward flowend)
```

 **Note:** The number 6 is the protocol (TCP).

The rule in FTD LINA:

```
<#root>
firepower#
show access-list | i 268438858
access-list CSM_FW_ACL_ line 17 remark rule-id 268438858: ACCESS POLICY: ACP1 - Mandatory
access-list CSM_FW_ACL_ line 18 remark rule-id 268438858: L7 RULE: trust_L3-L4
access-list CSM_FW_ACL_ line 19 advanced permit tcp object-group FMC_INLINE_src_rule_268438858 object-g
  access-list CSM_FW_ACL_ line 19 advanced
permit
  tcp host 192.168.10.50 host 192.168.11.50 eq www rule-id 268438858 (hitcnt=19) 0x9d442895
  access-list CSM_FW_ACL_ line 19 advanced
permit
  tcp host 192.168.10.50 host 192.168.11.51 eq www rule-id 268438858 (hitcnt=0) 0xd026252b
  access-list CSM_FW_ACL_ line 19 advanced
permit
  tcp host 192.168.10.51 host 192.168.11.50 eq www rule-id 268438858 (hitcnt=0) 0x0d785cc4
  access-list CSM_FW_ACL_ line 19 advanced
```

permit

```
tcp host 192.168.10.51 host 192.168.11.51 eq www rule-id 268438858 (hitcnt=0) 0x3b3234f1
```

## Verification:

Enable **system support trace** and initiate an HTTP session from host-A (192.168.10.50) to host-B (192.168.11.50). There are 3 packets forwarded to the Snort engine. Snort engine sends to LINA the PERMITLIST verdict which essentially offloads the rest of the flow to the LINA engine:

```
<#root>
```

```
>
```

```
system support trace
```

```
Enable firewall-engine-debug too? [n]:
```

```
y
```

```
Please specify an IP protocol:
```

```
tcp
```

```
Please
```

```
specify a client IP address:
```

```
192.168.10.50
```

```
Please specify a client port:
```

```
Please specify a server IP address:
```

```
192.168.11.50
```

```
Please specify a server port:
```

```
80
```

```
Monitoring packet tracer and firewall debug messages
```

```
192.168.10.50-42126 - 192.168.11.50-80 6 AS 1-1 CID 0 Packet:
```

```
TCP, SYN
```

```
, seq 453426648
```

```
192.168.10.50-42126 - 192.168.11.50-80 6 AS 1-1 CID 0 Session: new snort session
```

```
192.168.10.50-42126 - 192.168.11.50-80 6 AS 1-1 CID 0 AppID: service unknown (0), application unknown (
```

```
192.168.10.50-42126 > 192.168.11.50-80 6 AS 1-1 I 2 new firewall session
```

```
192.168.10.50-42126 > 192.168.11.50-80 6 AS 1-1 I 2 using HW or preset rule order 5, 'trust_L3-L4', act
```

```
192.168.10.50-42126 > 192.168.11.50-80 6 AS 1-1 I 2 HitCount data sent for rule id: 268438858,
```

```
192.168.10.50-42126 - 192.168.11.50-80 6 AS 1-1 CID 0 Firewall: trust/fastpath rule, 'trust_L3-L4', all
```

```
192.168.10.50-42126 - 192.168.11.50-80 6 AS 1-1 CID 0 Snort id 2, NAP id 2, IPS id 0,
```

```
Verdict PASS
```

```
192.168.11.50-80 - 192.168.10.50-42126 6 AS 1-1 CID 0 Packet:
```

```
TCP, SYN, ACK
```



```
, seq 2820426532, ack 453426649
192.168.11.50-80 - 192.168.10.50-42126 6 AS 1-1 CID 0 AppID: service unknown (0), application unknown (0)
192.168.11.50-80 - 192.168.10.50-42126 6 AS 1-1 CID 0 Firewall: trust/fastpath rule, 'trust_L3-L4', all
192.168.11.50-80 - 192.168.10.50-42126 6 AS 1-1 CID 0 Snort id 2, NAP id 2, IPS id 0,
```

**Verdict PASS**

```
192.168.10.50-42126 - 192.168.11.50-80 6 AS 1-1 CID 0 Packet:
```

**TCP, ACK**

```
, seq 453426649, ack 2820426533
192.168.10.50-42126 - 192.168.11.50-80 6 AS 1-1 CID 0 AppID: service unknown (0), application unknown (0)
192.168.10.50-42126 - 192.168.11.50-80 6 AS 1-1 CID 0 Firewall: trust/fastpath rule, 'trust_L3-L4', all
192.168.10.50-42126 - 192.168.11.50-80 6 AS 1-1 CID 0 Snort id 2, NAP id 2, IPS id 0,
```

**Verdict PERMITLIST**

Once the connection is terminated the Snort engine gets the metadata info from the LINA engine and deletes the session:

```
<#root>
```

```
192.168.10.50-42126 > 192.168.11.50-80 6 AS 1-1 I 2
```

**Got end of flow event**

```
from hardware with flags 00010001. Rule Match Data: rule_id 0, rule_action 0 rev_id 0, rule_flags 3
192.168.10.50-42126 > 192.168.11.50-80 6 AS 1-1 I 2 Logging EOF for event from hardware with rule_id = 0
192.168.10.50-42126 > 192.168.11.50-80 6 AS 1-1 I 2 :
```

**Received EOF, deleting the snort session.**

```
192.168.10.50-42126 - 192.168.11.50-80 6 AS 1-1 CID 0 Session: deleting snort session, reason: timeout
192.168.10.50-42126 > 192.168.11.50-80 6 AS 1-1 I 2 deleting firewall session flags = 0x10003, fwFlags = 0
192.168.10.50-42126 - 192.168.11.50-80 6 AS 1-1 CID 0 Session: deleted snort session using 0 bytes; pro
```

Snort capture shows the 3 packets that go to the Snort engine:

```
<#root>
```

```
>
```

**capture-traffic**

Please choose domain to capture traffic from:

- 0 - management0
- 1 - management1
- 2 - Global

Selection?

2

Please specify tcpdump options desired.  
(or enter '?' for a list of supported options)  
Options:

```
-n vlan and (host 192.168.10.50 and host 192.168.11.50)
```

```
10:26:16.525928 IP 192.168.10.50.42144 > 192.168.11.50.80: Flags [S], seq 3065553465, win 29200, option
10:26:16.525928 IP 192.168.11.50.80 > 192.168.10.50.42144: Flags [S.], seq 3581351172, ack 3065553466, w
10:26:16.525928 IP 192.168.10.50.42144 > 192.168.11.50.80: Flags [.] , ack 1, win 229, options [nop,nop,
```

LINA capture shows the flow which goes through it:

```
<#root>
```

```
firepower#
```

```
show capture CAPI
```

```
437 packets captured
```

```
1: 09:51:19.431007 802.1Q vlan#202 PO 192.168.10.50.42118 > 192.168.11.50.80: S 2459891187:24598911
2: 09:51:19.431648 802.1Q vlan#202 PO 192.168.11.50.80 > 192.168.10.50.42118: S 2860907367:28609073
3: 09:51:19.431847 802.1Q vlan#202 PO 192.168.10.50.42118 > 192.168.11.50.80: . ack 2860907368 win
4: 09:51:19.431953 802.1Q vlan#202 PO 192.168.10.50.42118 > 192.168.11.50.80: P 2459891188:24598913
5: 09:51:19.444816 802.1Q vlan#202 PO 192.168.11.50.80 > 192.168.10.50.42118: . 2860907368:28609087
6: 09:51:19.444831 802.1Q vlan#202 PO 192.168.11.50.80 > 192.168.10.50.42118: . 2860908736:28609101
```

```
...
```

Trace of the packets from LINA is another way to see the Snort verdicts. The first packet got the PASS verdict:

```
<#root>
```

```
firepower#
```

```
show capture CAPI packet-number 1 trace | i Type|Verdict
```

```
Type: CAPTURE
Type: ACCESS-LIST
Type: ROUTE-LOOKUP
Type: ACCESS-LIST
Type: CONN-SETTINGS
Type: NAT
Type: NAT
Type: IP-OPTIONS
Type: CAPTURE
Type: CAPTURE
Type: NAT
Type: CAPTURE
Type: NAT
Type: IP-OPTIONS
Type: CAPTURE
Type: FLOW-CREATION
Type: EXTERNAL-INSPECT
```

Type: SNORT

Snort id 22, NAP id 2, IPS id 0, Verdict PASS

Snort Verdict: (pass-packet) allow this packet

Type: INPUT-ROUTE-LOOKUP-FROM-OUTPUT-ROUTE-LOOKUP

Type: ADJACENCY-LOOKUP

Type: CAPTURE

Trace of the TCP SYN/ACK packet on the OUTSIDE interface:

<#root>

firepower#

show capture CAPO packet-number 2 trace | i Type|Verdict

Type: CAPTURE

Type: ACCESS-LIST

Type: FLOW-LOOKUP

Type: EXTERNAL-INSPECT

Type: SNORT

Snort id 22, NAP id 2, IPS id 0, Verdict PASS

Snort Verdict: (pass-packet) allow this packet

Type: INPUT-ROUTE-LOOKUP-FROM-OUTPUT-ROUTE-LOOKUP

Type: ADJACENCY-LOOKUP

Type: CAPTURE

The TCP ACK gets the PERMITLIST verdict:

<#root>

firepower#

show capture CAPI packet-number 3 trace | i Type|Verdict

Type: CAPTURE

Type: ACCESS-LIST

Type: FLOW-LOOKUP

Type: EXTERNAL-INSPECT

Type: SNORT

Snort id 22, NAP id 2, IPS id 0, Verdict PERMITLIST

Snort Verdict: (fast-forward) fast forward this flow

Type: CAPTURE

This is the full output from the Snort Verdict (packet #3)

<#root>

firepower#

```
show capture CAPI packet-number 3 trace | b Type: SNORT
```

```
Type: SNORT
```

```
Subtype:
```

```
Result: ALLOW
```

```
Config:
```

```
Additional Information:
```

```
Snort Trace:
```

```
Packet: TCP, ACK, seq 687485179, ack 1029625865
```

```
AppID: service unknown (0), application unknown (0)
```

```
Firewall: trust/fastpath rule, id 268438858, allow
```

```
Snort id 31, NAP id 2, IPS id 0,
```

```
Verdict PERMITLIST
```

```
Snort Verdict: (fast-forward) fast forward this flow
```

The 4th packet is not forwarded to the Snort engine since the verdict is cached by the LINA engine:

```
<#root>
```

```
firepower#
```

```
show capture CAPI packet-number 4 trace
```

```
441 packets captured
```

```
4: 10:34:02.741523      802.1Q vlan#202 P0 192.168.10.50.42158 > 192.168.11.50.80: P 164375589:1643
```

```
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: FLOW-LOOKUP
```

```
Subtype:
```

```
Result: ALLOW
```

```
Config:
```

```
Additional Information:
```

```
Found flow with id 1254, using existing flow
```

```
Phase: 4
```

```
Type: SNORT
```

```
Subtype:
```

```
Result: ALLOW
```

**Config:**  
**Additional Information:**  
Snort Verdict: (fast-forward) fast forward this flow

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

1 packet shown

Snort statistics confirm this:

<#root>

firepower#

show snort statistics

Packet Counters:

<b>Passed Packets</b>	<b>2</b>
Blocked Packets	0
Injected Packets	0
Packets bypassed (Snort Down)	0
Packets bypassed (Snort Busy)	0

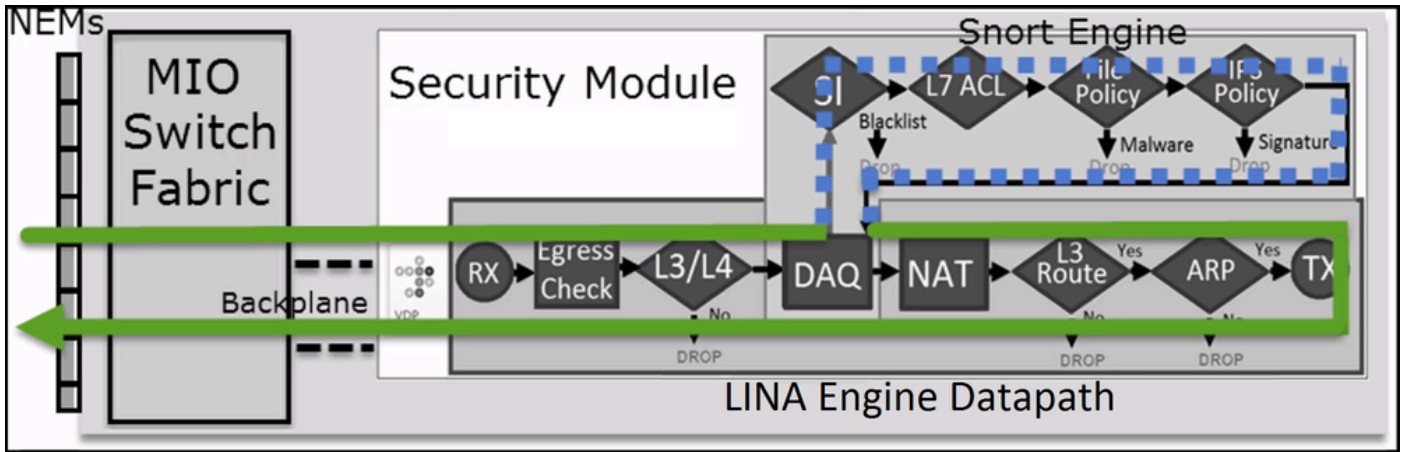
Flow Counters:

<b>Fast-Forwarded Flows</b>	<b>1</b>
Blacklisted Flows	0

Miscellaneous Counters:

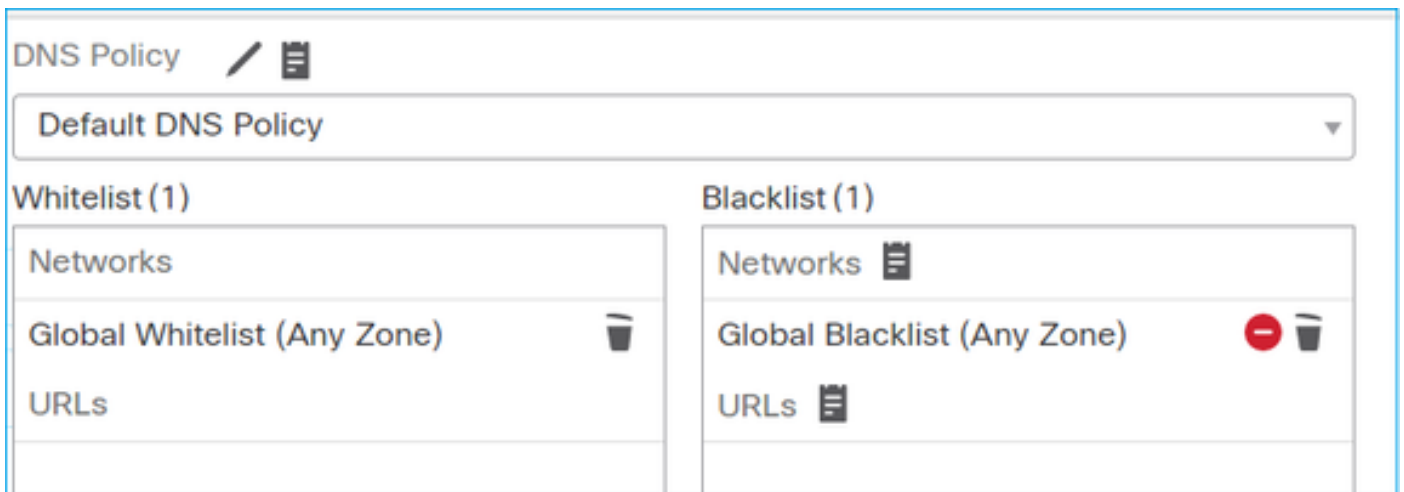
Start-of-Flow events	0
End-of-Flow events	1
Denied flow events	0
Frames forwarded to Snort before drop	0
Inject packets dropped	0

Packet flow with Trust rule. A few packets are inspected by Snort and the rest are inspected by LINA:



## Scenario 2. ACP Trust Action (without SI, QoS, and Identity Policy)

In case you want the FTD to apply Security Intelligence (SI) checks to all flows, SI is already enabled at the ACP level and you can specify the SI sources (TALOS, feeds, lists, etc). On the other hand, in case you want to disable it, you disable SI for Networks globally per ACP, SI for URL, and SI for DNS. The SI for Networks and URL is disabled as shown in the image:



In this case, the Trust rule is deployed to LINA as trust:

```
<#root>
```

```
>
```

```
show access-list
```

```
...
```

```
access-list CSM_FW_ACL_ line 9 remark rule-id 268435461: L4 RULE: Rule1
```

```
access-list CSM_FW_ACL_ line 10 advanced trust ip host 192.168.1.40 host 192.168.2.40 rule-id 268435461
```

 **Note:** As of 6.2.2 FTD supports TID. TID works in a way similar to SI, but in case SI is disabled, it does not 'force' packet redirection to the Snort engine for TID inspection.

## Verify the behavior

Initiate an HTTP session from host-A (192.168.1.40) to host-B (192.168.2.40). Since this is an FP4100 and supports Flow Offload in hardware these things happen:

- A few packets are forwarded through the FTD LINA engine and the rest of the flow is offloaded to SmartNIC (HW accelerator)
- No packets are forwarded to the Snort engine

The FTD LINA connection table shows the flag "o" which means the flow was offloaded to HW. Also, note the absence of the "N" flag. This essentially means 'no Snort redirection':

```
<#root>
firepower#
show conn
1 in use, 15 most used
TCP OUTSIDE 192.168.2.40:80 INSIDE 192.168.1.40:32809, idle 0:00:00, bytes 949584, flags UIO
o
```

Snort statistics show only logging events at the start and at the end of the session:

```
<#root>
firepower#
show snort statistics

Packet Counters:
  Passed Packets                0
  Blocked Packets               0
  Injected Packets              0
  Packets bypassed (Snort Down) 0
  Packets bypassed (Snort Busy) 0

Flow Counters:
  Fast-Forwarded Flows         0
  Blacklisted Flows            0

Miscellaneous Counters:
  start-of-Flow events         1
  End-of-Flow events           1
```

FTD LINA logs show that for each session there were 2 flows (one per each direction) offloaded to HW:

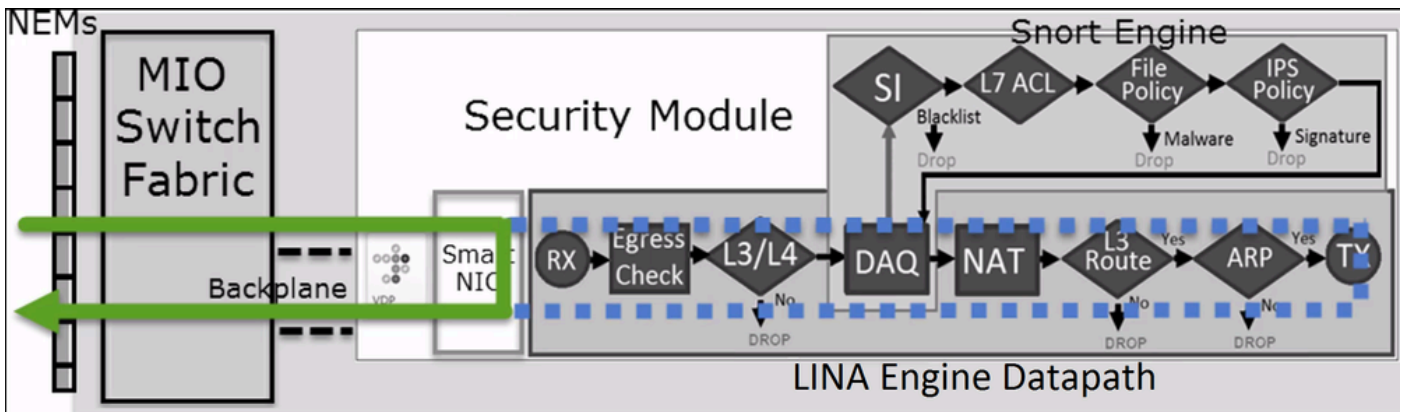
```
<#root>
Sep 27 2017 20:16:05: %ASA-7-609001: Built local-host INSIDE:192.168.1.40
Sep 27 2017 20:16:05: %ASA-6-302013: Built inbound TCP connection 25384 for INSIDE:192.168.1.40/32809 (
Sep 27 2017 20:16:05: %ASA-6-805001:
Offloaded TCP Flow for connection 25384 from INSIDE
```

```
:192.168.1.40/32809 (192.168.1.40/32809) to OUTSIDE:192.168.2.40/80 (192.168.2.40/80)
Sep 27 2017 20:16:05: %ASA-6-805001:
```

```
Offloaded TCP Flow for connection 25384 from OUTSIDE
```

```
:192.168.2.40/80 (192.168.2.40/80) to INSIDE:192.168.1.40/32809 (192.168.1.40/32809)
Sep 27 2017 20:16:05: %ASA-6-805002: TCP Flow is no longer offloaded for connection 25384 from OUTSIDE:
Sep 27 2017 20:16:05: %ASA-6-805002: TCP Flow is no longer offloaded for connection 25384 from INSIDE:
Sep 27 2017 20:16:05: %ASA-6-302014: Teardown TCP connection 25384 for INSIDE:192.168.1.40/32809 to OUT
Sep 27 2017 20:16:05: %ASA-7-609002: Teardown local-host INSIDE:192.168.1.40 duration 0:00:00
```

Packet flow with Trust rule deployed as **trust** action in LINA. A few packets are inspected by LINA and the rest are offloaded to SmartNIC (FP4100/FP9300):

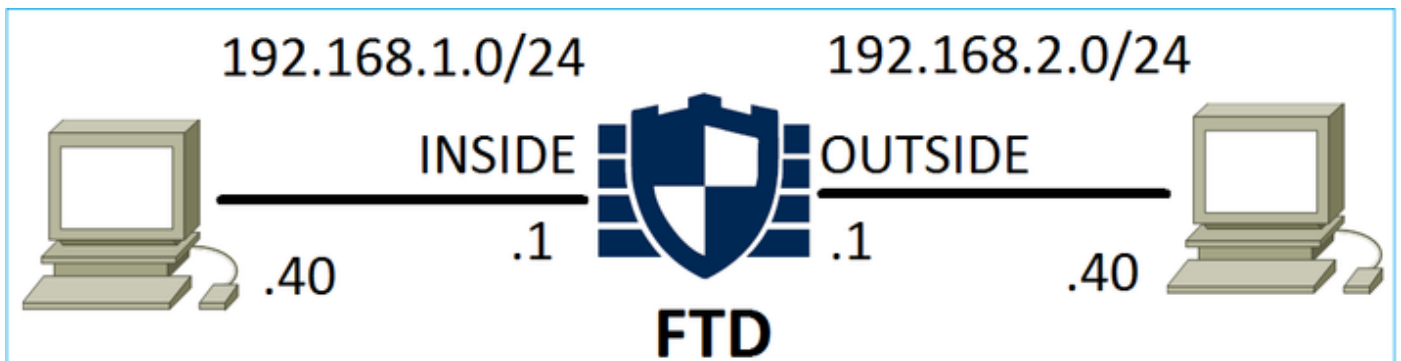


### Use Cases

- You must use **Trust** action when you want only a few packets to be checked by the Snort engine (for example Application detection, SI check) and the rest of the flow to be offloaded to the LINA engine
- If you use FTD on FP4100/9300 and want the flow to completely bypass the Snort inspection then consider the Prefilter rule with **Fastpath** action (see the related section in this document)

### Prefilter Policy Block Action

Consider the topology as shown in the image:



Consider also the policy as shown in the image:



Access Control ▸ Prefilter										
			Network Discovery		Application Detectors		Correlation		Actions ▾	
FTD_Prefilter										
Enter Description										
Rules										
<span>➕ Add Tunnel Rule</span> <span>➕ Add Prefilter Rule</span> <span>Search Rules</span>										
#	Name	Rule T...	...	De	Source	Destination	Source	Destinat...	VLAN Tag	Action
				Ini	Networks	Networks	Port	Port		
1	Prefilter1	Prefilter	any	any	192.168.1.40	192.168.2.40	any	any	any	✖ Block

This is the deployed policy in the FTD Snort engine (ngfw.rules file):

```
<#root>
# Start of tunnel and priority rules.
#
These rules are evaluated by LINA
. Only tunnel tags are used from the matched rule id.
268437506
deny
  any 192.168.1.40 32 any any 192.168.2.40 32 any any any (tunnel -1
```

In LINA:

```
<#root>
access-list CSM_FW_ACL_ line 1 remark rule-id 268437506: PREFILTER POLICY: FTD_Prefilter
access-list CSM_FW_ACL_ line 2 remark rule-id 268437506: RULE: Prefilter1
access-list CSM_FW_ACL_ line 3 advanced
deny
  ip host 192.168.1.40 host 192.168.2.40 rule-id 268437506 event-log flow-start (hitcnt=0) 0x76476240
```

When you trace a virtual packet, it shows that the packet is dropped by LINA and never forwarded to Snort:

```
<#root>
firepower# packet-tracer input INSIDE icmp 192.168.1.40 8 0 192.168.2.40
...
Phase: 4
Type: ACCESS-LIST
Subtype: log
Result: DROP
Config:
access-group CSM_FW_ACL_ global
```

```
access-list CSM_FW_ACL_ advanced deny ip host 192.168.1.40 host 192.168.2.40 rule-id 268437506 event-log
```

```
access-list CSM_FW_ACL_ remark rule-id 268437506:
```

```
PREFILTER POLICY: FTD_Prefilter
```

```
access-list CSM_FW_ACL_ remark rule-id 268437506: RULE: Prefilter1
```

```
Additional Information:
```

```
Result:
```

```
input-interface: INSIDE
```

```
input-status: up
```

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

```
output-interface: OUTSIDE
```

```
output-status: up
```

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

```
Action: drop
```

```
Drop-reason: (acl-drop) Flow is denied by configured rule
```

Snort statistics show:

```
<#root>
```

```
firepower#
```

```
show snort statistics
```

```
Packet Counters:
```

```
  Passed Packets                                0
```

```
  Blocked Packets                              0
```

```
  Injected Packets                             0
```

```
  Packets bypassed (Snort Down)                0
```

```
  Packets bypassed (Snort Busy)                0
```

```
Flow Counters:
```

```
  Fast-Forwarded Flows                         0
```

```
  Blacklisted Flows                            0
```

```
Miscellaneous Counters:
```

```
  Start-of-Flow events                         0
```

```
  End-of-Flow events                           0
```

```
  Denied flow events                           1
```

LINA ASP drops show:

```
<#root>
```

```
firepower#
```

```
show asp drop
```

```
Frame drop:
```

Flow is denied by configured rule (acl-drop)

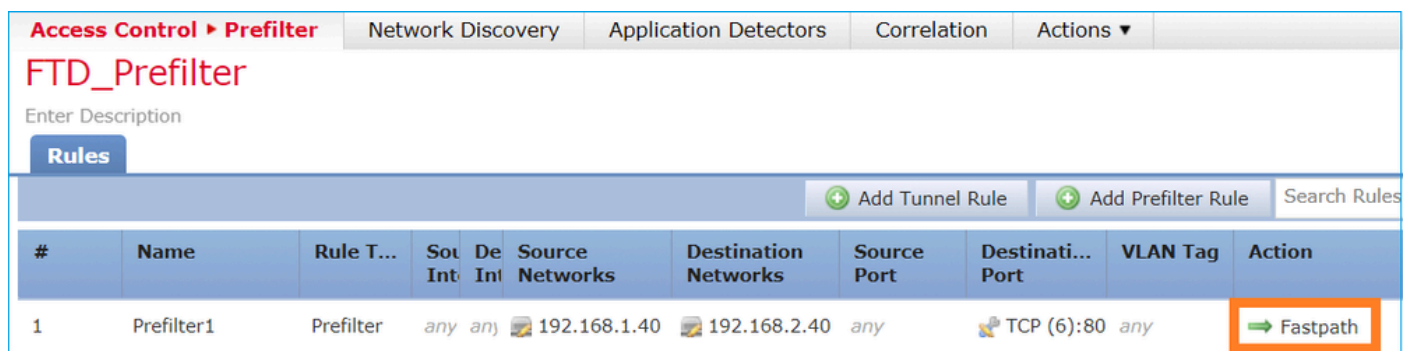
1

## Use Cases

You can use a Prefilter Block rule when you want to block traffic based on L3/L4 conditions and without the need to do any Snort inspection of the traffic.

## Prefilter Policy Fastpath Action

Consider the Prefilter Policy rule as shown in the image:



The screenshot shows the configuration page for a Prefilter rule named 'FTD\_Prefilter'. The 'Rules' tab is active, displaying a table of rules. The first rule, 'Prefilter1', is highlighted. Its configuration is as follows:

#	Name	Rule T...	Source Int	Dest Int	Source Networks	Destination Networks	Source Port	Destinati... Port	VLAN Tag	Action
1	Prefilter1	Prefilter	any	any	192.168.1.40	192.168.2.40	any	TCP (6):80	any	→ Fastpath

This is the deployed policy in the FTD Snort engine:

```
<#root>
```

```
268437506
```

```
fastpath
```

```
any any any any any any any any (log dcforward flowend) (tunnel -1)
```

In FTD LINA:

```
<#root>
```

```
access-list CSM_FW_ACL_ line 1 remark rule-id 268437506:
```

```
PREFILTER POLICY: FTD_Prefilter
```

```
access-list CSM_FW_ACL_ line 2 remark rule-id 268437506: RULE: Prefilter1
```

```
access-list CSM_FW_ACL_ line 3 advanced
```

```
trust
```

```
tcp host 192.168.1.40 host 192.168.2.40 eq www rule-id 268437506 event-log flow-end (hitcnt=0) 0xf3410
```

## Verify Behavior

When host-A (192.168.1.40) tries to open an HTTP session to host-B (192.168.2.40) a few packets go through LINA and the rest are offloaded to SmartNIC. In this case `system support trace` with `firewall-engine-debug` enabled shows:

```
<#root>
```

```
>
```

```
system support trace
```

```
Please specify an IP protocol: tcp
Please specify a client IP address: 192.168.1.40
Please specify a client port:
Please specify a server IP address: 192.168.2.40
Please specify a server port:
Enable firewall-engine-debug too? [n]: y
Monitoring packet tracer debug messages
```

```
192.168.1.40-32840 > 192.168.2.40-80 6 AS 1 I 8 Got end of flow event from hardware
with flags 04000000
```

LINA logs show the offloaded flow:

```
<#root>
```

```
Oct 01 2017 14:36:51: %ASA-7-609001: Built local-host INSIDE:192.168.1.40
Oct 01 2017 14:36:51: %ASA-7-609001: Built local-host OUTSIDE:192.168.2.40
Oct 01 2017 14:36:51: %ASA-6-302013: Built inbound TCP connection 966 for INSIDE:192.168.1.40/32840 (19
Oct 01 2017 14:36:51: %ASA-6-805001:
```

```
Offloaded TCP Flow for connection 966 from INSIDE
```

```
:192.168.1.40/32840 (192.168.1.40/32840) to OUTSIDE:192.168.2.40/80 (192.168.2.40/80)
Oct 01 2017 14:36:51: %ASA-6-805001:
```

```
Offloaded TCP Flow for connection 966 from OUTSIDE
```

```
:192.168.2.40/80 (192.168.2.40/80) to INSIDE:192.168.1.40/32840 (192.168.1.40/32840)
```

LINA captures show 8 packets go through:

```
<#root>
```

```
firepower# show capture
```

```
capture CAPI type raw-data buffer 33554432 trace trace-count 100 interface INSIDE [Capturing -
3908 bytes]
```

```
match ip host 192.168.1.40 host 192.168.2.40
```

```
capture CAPO type raw-data buffer 33554432 trace trace-count 100 interface OUTSIDE [Capturing -
3908 bytes]
```

```
match ip host 192.168.1.40 host 192.168.2.40
```

```
<#root>
```

```
firepower#
```

```
show capture CAPI
```

```
8 packets captured
```

```
1: 14:45:32.700021 192.168.1.40.32842 > 192.168.2.40.80: S 3195173118:3195173118(0) win 2920 <mss 1
2: 14:45:32.700372 192.168.2.40.80 > 192.168.1.40.32842: S 184794124:184794124(0) ack 3195173119 wi
3: 14:45:32.700540 192.168.1.40.32842 > 192.168.2.40.80: P 3195173119:3195173317(198) ack 184794125
4: 14:45:32.700876 192.168.2.40.80 > 192.168.1.40.32842: . 184794125:184795493(1368) ack 3195173317
5: 14:45:32.700922 192.168.2.40.80 > 192.168.1.40.32842: P 184795493:184796861(1368) ack 3195173317
6: 14:45:32.701425 192.168.2.40.80 > 192.168.1.40.32842: FP 184810541:184810851(310) ack 3195173317
7: 14:45:32.701532 192.168.1.40.32842 > 192.168.2.40.80: F 3195173317:3195173317(0) ack 184810852 w
8: 14:45:32.701639 192.168.2.40.80 > 192.168.1.40.32842: . ack 3195173318 win 2697 <nop,nop,timesta
```

FTD Flow-offload statistics show 22 packets offloaded to HW:

```
<#root>
```

```
firepower#
```

```
show flow-offload statistics
```

```
Packet stats of port : 0
```

```
Tx Packet count          :                22
```

```
Rx Packet count          :                22
```

```
Dropped Packet count    :                0
VNIC transmitted packet :                22
VNIC transmitted bytes  :             15308
VNIC Dropped packets    :                0
VNIC erroneous received :                0
VNIC CRC errors         :                0
VNIC transmit failed    :                0
VNIC multicast received :                0
```

You can also use the `show flow-offload flow` command to see additional information related to the offloaded flows. Here is an example:

```
<#root>
```

```
firepower#
```

```
show flow-offload flow
```

```
Total offloaded flow stats: 2 in use, 4 most used, 20% offloaded, 0 collisions
TCP intfc 103 src 192.168.1.40:39301 dest 192.168.2.40:20, static, timestamp 616063741, packets 33240,
TCP intfc 104 src 192.168.2.40:20 dest 192.168.1.40:39301, static, timestamp 616063760, packets 249140,
firepower#
```

```
show conn
```

```
5 in use, 5 most used
```

```
Inspect Snort:
```

```
    preserve-connection: 1 enabled, 0 in effect, 4 most enabled, 0 most in effect
```

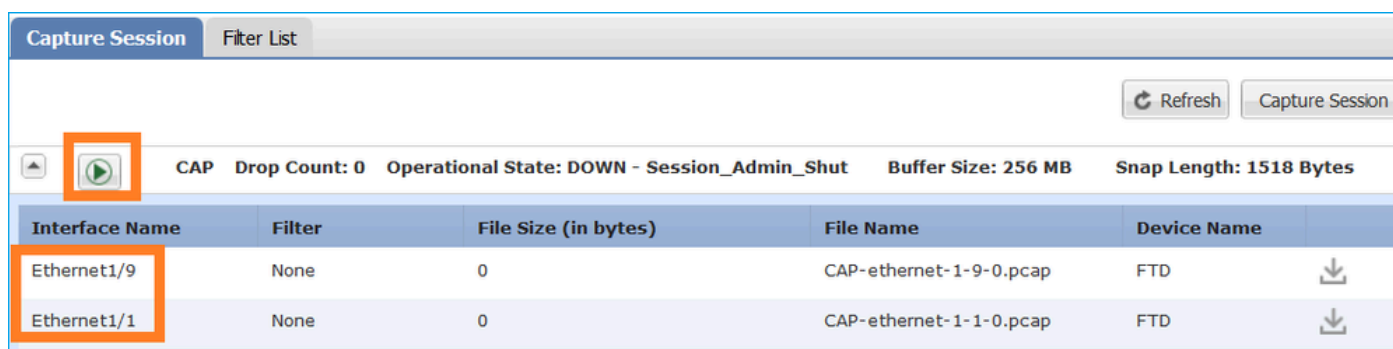
```
TCP OUTSIDE 192.168.2.40:21 INSIDE 192.168.1.40:40988, idle 0:00:00, bytes 723, flags UIO
TCP OUTSIDE 192.168.2.40:21 INSIDE 192.168.1.40:40980, idle 0:02:40, bytes 1086, flags UIO
TCP OUTSIDE 192.168.2.40:80 INSIDE 192.168.1.40:49442, idle 0:00:00, bytes 86348310, flags UIO N1
TCP OUTSIDE 192.168.2.40:20 INSIDE 192.168.1.40:39301, idle 0:00:00, bytes 485268628, flags U
```

```
o <- offloaded flow
```

```
TCP OUTSIDE 192.168.2.40:20 INSIDE 192.168.1.40:34713, idle 0:02:40, bytes 821799360, flags UFRIO
```

- The percentage is based on the "show conn" output. For example, if 5 conns in total go through the FTD LINA engine and 1 of them is offloaded then 20% is reported as offloaded
- The maximum limit of offloaded sessions depends on the software version (for example ASA 9.8.3 and FTD 6.2.3 support 4 million bi-directional (or 8 million unidirectional) offloaded flows)
- In case the number of offloaded flows reaches the limit (for example 4 million bi-directional flows) no new connections are offloaded until current connections are removed from the offloaded table

In order to see all the packets on FP4100/9300 that go through FTD (offloaded + LINA) there is a need to enable capture at chassis level as shown in the image:



Chassis backplane capture shows both directions. Due to FXOS capture architecture (2 capture points per direction) every packet is shown **twice** as shown in the image:

Packet statistics:

- Total packets through FTD: 30
- Packets through FTD LINA: 8
- Packets offloaded to SmartNIC HW accelerator: 22

In the case of a platform different than FP4100/FP9300 all the packets are handled by the LINA engine since flow-offload is not supported (note the absence of the o flag):

```
<#root>
```

```
FP2100-6#
```

```
show conn addr 192.168.1.40
```

```
33 in use, 123 most used
```

```
Inspect Snort:
```

```
    preserve-connection: 0 enabled, 0 in effect, 2 most enabled, 0 most in effect
```

```
TCP OUTSIDE 192.168.2.40:80 INSIDE 192.168.1.40:50890, idle 0:00:09, bytes 175, flags
```

```
UxIO
```

The LINA syslogs only show connection setup and connection termination events:

```
<#root>
```

```
FP2100-6#
```

```
show log | i 192.168.2.40
```

```
Jun 21 2020 14:29:44: %FTD-6-302013:
```

```
Built inbound TCP
```

```
    connection 6914 for INSIDE:192.168.1.40/50900 (192.168.11.101/50900) to OUTSIDE:192.168.2.40/80 (192.1
```

```
Jun 21 2020 14:30:30: %FTD-6-302014:
```

```
Teardown TCP connection
```

```
    6914 for INSIDE:192.168.1.40/50900 to OUTSIDE:192.168.2.40/80 duration 0:00:46 bytes 565 TCP FINs from
```

## Use Cases

- Use **Prefilter Fastpath** action when you want to bypass completely the Snort inspection. You typically want to do this for big fat flows that you trust like backups, database transfers, etc
- On FP4100/9300 appliances the **Fastpath** action triggers flow-offload and only a few packets go through the FTD LINA engine. The rest is handled by SmartNIC which decreases the latency

## Prefilter Policy Fastpath Action (Inline-Set)

In case a Prefilter Policy Fastpath action is applied on traffic that goes through an inline-set (NGIPS interfaces) these points must be taken into consideration:

- The rule is applied to the LINA engine as a **trust** action
- The flow is not inspected by the Snort engine
- Flow offload (HW acceleration) does not occur since flow offload is not applicable on NGIPS interfaces

Here is an example of a packet trace in the case of Prefilter Fastpath action applied on an inline-set:

```
firepower# packet-tracer input inside tcp 192.168.1.40 12345 192.168.1.50 80 detailed
```

```
Phase: 1
```

```
Type: NGIPS-MODE
```

Subtype: ngips-mode

Result: ALLOW

Config:

Additional Information:

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

Forward Flow based lookup yields rule:

in id=0x2ad7ac48b330, priority=501, domain=ips-mode, deny=false  
hits=2, user\_data=0x2ad80d54abd0, cs\_id=0x0, 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  
input\_ifc=inside, output\_ifc=any

Phase: 2

Type: ACCESS-LIST

Subtype: log

Result: ALLOW

Config:

access-group CSM\_FW\_ACL\_ global

access-list CSM\_FW\_ACL\_ advanced trust ip object 192.168.1.0 object 192.168.1.0 rule-id 268438531 event

access-list CSM\_FW\_ACL\_ remark rule-id 268438531: PREFILTER POLICY: PF1

access-list CSM\_FW\_ACL\_ remark rule-id 268438531: RULE: 1

Additional Information:

Forward Flow based lookup yields rule:

in id=0x2ad9f9f8a7f0, priority=12, domain=permit, trust  
hits=1, user\_data=0x2ad9b23c5d40, cs\_id=0x0, use\_real\_addr, flags=0x0, protocol=0  
src ip/id=192.168.1.0, mask=255.255.255.0, port=0, tag=any, ifc=any  
dst ip/id=192.168.1.0, mask=255.255.255.0, port=0, tag=any, ifc=any, vlan=0, dscp=0x0  
input\_ifc=any, output\_ifc=any

Phase: 3

Type: NGIPS-EGRESS-INTERFACE-LOOKUP

Subtype: Resolve Egress Interface

Result: ALLOW

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

Config:

Additional Information:

New flow created with id 7, packet dispatched to next module

Module information for forward flow ...

snp\_fp\_ips\_tcp\_state\_track\_lite

snp\_fp\_ips\_mode\_adj

snp\_fp\_tracer\_drop

snp\_ifc\_stat

Module information for reverse flow ...

snp\_fp\_ips\_tcp\_state\_track\_lite

snp\_fp\_ips\_mode\_adj

snp\_fp\_tracer\_drop

snp\_ifc\_stat

Result:

input-interface: inside

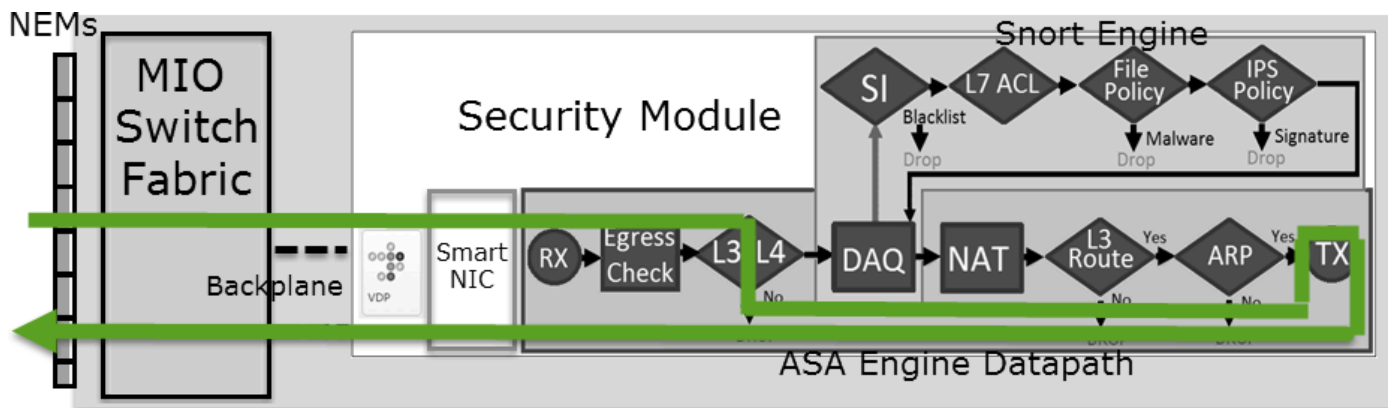
input-status: up

input-line-status: up

Action: allow



This is the visual representation of the packet path:



### Prefilter Policy Fastpath Action (Inline-Set with Tap)

Same as the Inline-Set case

### Prefilter Policy Analyze Action

#### Scenario 1. Prefilter Analyze with ACP Block Rule

Consider the Prefilter Policy which contains an Analyze rule as shown in the image:

Access Control ▸ Prefilter										
			Network Discovery		Application Detectors		Correlation		Actions ▾	
Prefilter_Policy1										
Enter Description										
Rules										
<div style="text-align: right;"> <span>➕ Add Tunnel Rule</span> <span>➕ Add Prefilter Rule</span> <span>Search R</span> </div>										
#	Name	Rule T...	Source Interfac...	Destinat... Interfac...	Source Networks	Destination Networks	Source Port	Destinat... Port	VLAN Tag	Action
1	Prefilter_Rule1	Prefilter	any	any	🌐 192.168.1.40	🌐 192.168.2.40	any	any	any	✔ Analyze

The ACP contains only the default rule which is set to Block All Traffic as shown in the image:

Access Control > Access Control | Network Discovery | Application Detectors | Correlation | Actions ▾

**ACP1**  
Enter Description

Prefilter Policy: **Prefilter\_Policy1** | SSL Policy: None

Inheritance S

Rules | Security Intelligence | HTTP Responses | Advanced

Show Rule Conflicts

#	Name	Source Zones	Dest Zones	Source Netwo...	Dest Netwo...	VLAN ...	Users	Applic...	Sourc...	Dest P...	URLs	ISE/S... Attrib...	Action
▼ Mandatory - ACP1 (-)													
There are no rules in this section. <a href="#">Add Rule</a> or <a href="#">Add Category</a>													
▼ Default - ACP1 (-)													
There are no rules in this section. <a href="#">Add Rule</a> or <a href="#">Add Category</a>													
Default Action												Access Control: Block All Traffic	

This is the deployed policy in the FTD Snort engine (ngfw.rules file):

```
<#root>
# Start of tunnel and priority rules.
# These rules are evaluated by LINA. Only tunnel tags are used from the matched rule id.
268435460 allow any 192.168.1.40 32 any any 192.168.2.40 32 any any any (tunnel -1)
268435459 allow any any 1025-65535 any any 3544 any 17 (tunnel -1)
268435459 allow any any 3544 any any 1025-65535 any 17 (tunnel -1)
268435459 allow any any any any any any any any 47 (tunnel -1)
268435459 allow any any any any any any any any 41 (tunnel -1)
268435459 allow any any any any any any any any 4 (tunnel -1)
# End of tunnel and priority rules.
# Start of AC rule.
268435458 deny any any any any any any any any any (log dcfoward flowstart)
# End of AC rule.
```


This is the deployed policy in FTD LINA engine:

```
<#root>
access-list CSM_FW_ACL_ line 3 advanced
permit
ip host 192.168.1.40 host 192.168.2.40 rule-id 268435460 (hitcnt=0) 0xb788b786
```

## Verify Behavior

Packet-tracer shows that the packet is allowed by LINA, is forwarded to Snort engine (due to **permit** action) and Snort Engine returns a **Block** verdict since the default action from AC is matched.

---

 **Note:** Snort does not evaluate traffic based on tunnel rules

---

When you trace a packet it reveals the same:

<#root>

firepower#

packet-tracer input INSIDE icmp 192.168.1.40 8 0 192.168.2.40

...

Phase: 4

Type: ACCESS-LIST

Subtype: log

Result: ALLOW

Config:

access-group CSM\_FW\_ACL\_ global

access-list CSM\_FW\_ACL\_ advanced permit ip host 192.168.1.40 host 192.168.2.40 rule-id 268435460

access-list CSM\_FW\_ACL\_ remark rule-id 268435460: PREFILTER POLICY: Prefilter\_Policy1

access-list CSM\_FW\_ACL\_ remark rule-id 268435460: RULE: Prefilter\_Rule1

Additional Information:

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

...

Phase: 14

Type: SNORT

Subtype:

Result: DROP

Config:

Additional Information:

Snort Trace:

Packet: ICMP

AppID: service ICMP (3501), application unknown (0)

Firewall: starting rule matching, zone -1 -> -1, geo 0 -> 0, vlan 0, sgt 65535, user 9999997, icmpType

Firewall: block rule, id 268435458, drop

Snort: processed decoder alerts or actions queue, drop

NAP id 1, IPS id 0,

Verdict BLOCKLIST, Blocked by Firewall

Snort Verdict:

(block-list) block list this flow

Result:

input-interface: INSIDE

input-status: up

input-line-status: up

output-interface: OUTSIDE

output-status: up

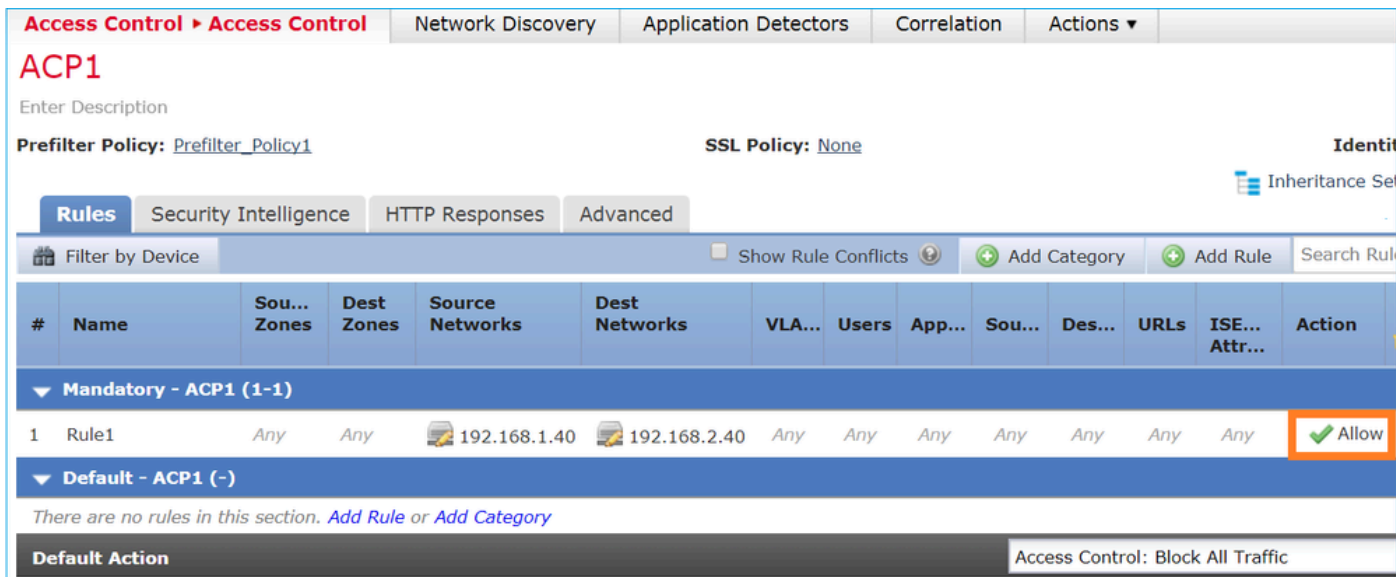
output-line-status: up

Action: drop

Drop-reason: (firewall) Blocked by the firewall preprocessor

## Scenario 2. Prefilter Analyze with ACP Allow Rule

If the goal is to allow the packet to traverse through the FTD, there is a need to add a rule in ACP. The Action can be either Allow or Trust which depends on the goal (for example if you want to apply an L7 inspection you must use Allow action) as shown in the image:



The deployed policy in FTD Snort engine:

```
<#root>
```

```
# Start of AC rule.
```

```
268435461 allow any 192.168.1.40 32 any any 192.168.2.40 32 any any any
```

```
268435458 deny any any any any any any any any (log dcforward flowstart)
```

```
# End of AC rule.
```

In LINA engine:

```
<#root>
```

```
access-list CSM_FW_ACL_ line 3 advanced
```

```
permit
```

```
ip host 192.168.1.40 host 192.168.2.40 rule-id 268435460 (hitcnt=1) 0xb788b786
```

## Verify Behavior

Packet-tracer shows that the packet matches rule 268435460 in LINA and 268435461 in Snort engine:

```
<#root>
```

firepower#

```
packet-tracer input INSIDE icmp 192.168.1.40 8 0 192.168.2.40
```

```
...
Phase: 4
Type: ACCESS-LIST
Subtype: log
Result: ALLOW
Config:
access-group CSM_FW_ACL_ global
access-list CSM_FW_ACL_ advanced

permit

ip host 192.168.1.40 host 192.168.2.40 rule-id 268435460
access-list CSM_FW_ACL_ remark rule-id 268435460: PREFILTER POLICY: Prefilter_Policy1
access-list CSM_FW_ACL_ remark rule-id 268435460: RULE: Prefilter_Rule1
Additional Information:
```

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

```
...
Phase: 14
Type: SNORT
Subtype:
Result: ALLOW
Config:
Additional Information:
Snort Trace:
Packet: ICMP
AppID: service ICMP (3501), application unknown (0)
Firewall: starting rule matching, zone -1 -> -1, geo 0 -> 0, vlan 0, sgt 65535, user 9999997, icmpType
Firewall:

allow rule, id 268435461, allow

NAP id 1, IPS id 0,

Verdict PASS

Snort Verdict: (pass-packet) allow this packet
```

```
...
Result:
input-interface: INSIDE
input-status: up
input-line-status: up
output-interface: OUTSIDE
output-status: up
output-line-status: up

Action: allow
```

### Scenario 3. Prefilter Analyze with ACP Trust Rule

In case the ACP contains a Trust rule then you have this as shown in the image:

Access Control ▸ Access Control    Network Discovery    Application Detectors    Correlation    Actions ▾

## ACP1

Enter Description

Prefilter Policy: [Prefilter\\_Policy1](#)      SSL Policy: [None](#)      Identif...

Inheritance Se...

Rules    Security Intelligence    HTTP Responses    Advanced

Filter by Device    Show Rule Conflicts    Add Category    Add Rule    Search Rule

#	Name	Sou... Zones	Dest Zones	Source Networks	Dest Networks	VLA...	Users	App...	Sou...	Des...	URLs	ISE... Attr...	Action
▼ Mandatory - ACP1 (1-1)													
1	Rule1	Any	Any	192.168.1.40	192.168.2.40	Any	Any	Any	Any	Any	Any	Any	→ Trust
▼ Default - ACP1 (-)													
There are no rules in this section. <a href="#">Add Rule</a> or <a href="#">Add Category</a>													
Default Action											Access Control: Block All Traffic		

Snort:

```
<#root>
```

```
# Start of AC rule.
```

```
268435461 fastpath any 192.168.1.40 32 any any 192.168.2.40 32 any any any
```

```
268435458 deny any any any any any any any any (log dcforward flowstart)
```

```
# End of AC rule.
```

LINA:

```
<#root>
```

```
access-list CSM_FW_ACL_ line 3 advanced
```

```
permit
```

```
ip host 192.168.1.40 host 192.168.2.40 rule-id 268435460 (hitcnt=2) 0xb788b786
```

Remember that since the SI is enabled by default, the Trust rule is deployed as **permit** action on LINA so at least a few packets are redirected to the Snort engine for inspection.

### Verify Behavior

Packet-tracer shows that the Snort engine Permitlists the packet and essentially offloads the rest flow to LINA:

```
<#root>
```

```
firepower#
```

```
packet-tracer input INSIDE icmp 192.168.1.40 8 0 192.168.2.40
```

```
...
```

```
Phase: 4
```

```

Type: ACCESS-LIST
Subtype: log
Result: ALLOW
Config:
access-group CSM_FW_ACL_ global

access-list CSM_FW_ACL_ advanced permit ip host 192.168.1.40 host 192.168.2.40 rule-id 268435460

access-list CSM_FW_ACL_ remark rule-id 268435460: PREFILTER POLICY: Prefilter_Policy1
access-list CSM_FW_ACL_ remark rule-id 268435460: RULE: Prefilter_Rule1
Additional Information:

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

...
Phase: 14
Type: SNORT
Subtype:
Result: ALLOW
Config:
Additional Information:
Snort Trace:
Packet: ICMP
AppID: service ICMP (3501), application unknown (0)
Firewall: starting rule matching, zone -1 -> -1, geo 0 -> 0, vlan 0, sgt 65535, user 9999997, icmpType
Firewall:

trust/fastpath rule, id 268435461, allow

NAP id 1, IPS id 0,

  Verdict PERMITLIST

Snort Verdict: (fast-forward) fast forward this flow

...
Result:
input-interface: INSIDE
input-status: up
input-line-status: up
output-interface: OUTSIDE
output-status: up
output-line-status: up

Action: allow

```

#### Scenario 4. Prefilter Analyze with ACP Trust Rule

In this scenario the SI was disabled manually.

The rule is deployed in Snort as follows:

```

<#root>

# Start of AC rule.
268435461

fastpath

  any 192.168.1.40 32 any any 192.168.2.40 32 any any any
268435458 deny any any any any any any any (log dcforward flowstart)

```

```
# End of AC rule.
```

In LINA the rule is deployed as trust. A packet though matches the permit rule (see the ACE hit counts) that is deployed due to Analyze Prefilter rule and the packet is inspected by the Snort engine:

```
<#root>
access-list CSM_FW_ACL_ line 3 advanced
permit ip host 192.168.1.40 host 192.168.2.40
  rule-id 268435460 (hitcnt=3) 0xb788b786
...
access-list CSM_FW_ACL_ line 13 advanced
trust ip host 192.168.1.40 host 192.168.2.40
  rule-id 268435461 event-log flow-end (hitcnt=0) 0x5c1346d6
...
access-list CSM_FW_ACL_ line 16 advanced
deny ip any any
  rule-id 268435458 event-log flow-start (hitcnt=0) 0x97aa021a
```

## Verify Behavior

```
<#root>
firepower#
packet-tracer input INSIDE icmp 192.168.1.40 8 0 192.168.2.40
...
Phase: 4
Type: ACCESS-LIST
Subtype: log
Result: ALLOW
Config:
access-group CSM_FW_ACL_ global
access-list CSM_FW_ACL_ advanced
permit
  ip host 192.168.1.40 host 192.168.2.40 rule-id 268435460
access-list CSM_FW_ACL_ remark rule-id 268435460: PREFILTER POLICY: Prefilter_Policy1
access-list CSM_FW_ACL_ remark rule-id 268435460: RULE: Prefilter_Rule1
Additional Information:
```

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

```
...
Phase: 14
Type: SNORT
Subtype:
Result: ALLOW
Config:
Additional Information:
```



```

Snort Trace:
Packet: ICMP
AppID: service ICMP (3501), application unknown (0)
Firewall: starting rule matching, zone -1 -> -1, geo 0 -> 0, vlan 0, sgt 65535, user 9999997, icmpType
Firewall:

trust/fastpath rule, id 268435461, allow

NAP id 1, IPS id 0,

Verdict PERMITLIST

Snort Verdict: (fast-forward) fast forward this flow
...
Result:
input-interface: INSIDE
input-status: up
input-line-status: up
output-interface: OUTSIDE
output-status: up
output-line-status: up

Action: allow

```

## Main Points

- The **Analyze** Action is deployed as a permit rule in the LINA engine. This has an effect on the packet to be forwarded to the Snort engine for inspection
- The **Analyze** Action does not deploy any rule in the Snort engine so you need to ensure that you configure a rule in ACP that is matched in Snort<
- It depends on the ACP rule that is deployed in the Snort engine (**block** vs **allow** vs **fastpath**) none or all or a few packets are allowed by Snort

## Use Cases

- A use case of **Analyze** Action is when you have a broad **Fastpath** rule in the Prefilter policy and you want to put some exceptions for specific flows so that they are inspected by Snort

## ACP Monitor Action

A monitor rule configured on FMC UI:

Name	Sou... Zones	Dest Zones	Source Networks	Dest Networks	VLAN Tags	Users	Applic...	Sou... Ports	Dest Ports	URLs	Sou... SGT	Dest SGT	Action	Icons
1 Monitor_Rule	Any	Any	192.168.10.0/24	192.168.11.0/24	Any	Any	Any	Any	Any	Any	Any	Any	Monitor	🛡️ 📄 🔄 🗑️

The monitor rule is deployed on the FTD LINA engine as a **permit** action and to the Snort engine as an **audit** action.

<#root>

```

firepower#
show access-list
...
access-list CSM_FW_ACL_ line 10 advanced
permit
ip 192.168.10.0 255.255.255.0 192.168.11.0 255.255.255.0 rule-id
268438863
(hitcnt=0) 0x61bbaf0c

```

The Snort rule:

```

<#root>
admin@firepower:~$
cat /var/sf/detection_engines/9e080e5c-adc3-11ea-9d37-44884cf7e9ba/ngfw.rules
...
# Start of AC
rule.
268438863 audit
any 192.168.10.0 24 any any 192.168.11.0 24 any any any (log dcforward flowend)
# End rule 268438863

```

## Main Points

- Monitor Rule does not drop or permit traffic but generates a Connection Event. The packet is checked against subsequent rules and it is either allowed or dropped
- FMC Connection Events show that the packet matched 2 rules:

Connection Events [\(switch workflow\)](#)

No Search Constraints [\(Edit Search\)](#)

Connections with Application Details [Table View of Connection Events](#)

Jump to...

	<input type="checkbox"/>	First Packet ×	Last Packet ×	Action ×	Initiator IP ×	Responder IP ×	Source Port / ICMP Type ×	Destination Port / ICMP Code ×	Access Control Policy ×	Access Control Rule ×
▼	<input type="checkbox"/>	2020-06-20 22:17:40	2020-06-20 22:17:43	Trust	192.168.10.50	192.168.11.50	41920 / tcp	80 (http) / tcp	ACP1	trust_L3-L4, Monitor_Rule

System support trace output shows that packets match both rules:

```

<#root>
>
system support trace

```

Enable firewall-engine-debug too? [n]:

y

Please specify an IP protocol: tcp

Please specify a client IP address:

192.168.10.50

Please specify a client port:

Please specify a server IP address:

192.168.11.50

Please specify a server port:

Monitoring packet tracer and firewall debug messages

192.168.10.50-41922 - 192.168.11.50-80 6 AS 1-1 CID 0 Packet: TCP, SYN, seq 419031630

192.168.10.50-41922 - 192.168.11.50-80 6 AS 1-1 CID 0 Session: new snort session

192.168.10.50-41922 - 192.168.11.50-80 6 AS 1-1 CID 0 AppID: service unknown (0), application unknown (0)

192.168.10.50-41922 > 192.168.11.50-80 6 AS 1-1 I 19 new firewall session

192.168.10.50-41922 > 192.168.11.50-80 6 AS 1-1 I 19 Starting AC with minimum 2, 'Monitor\_Rule', and IP

192.168.10.50-41922 > 192.168.11.50-80 6 AS 1-1 I 19

match rule order 2, 'Monitor\_Rule', action Audit

192.168.10.50-41922 > 192.168.11.50-80 6 AS 1-1 I 19

match rule order 3, 'trust\_L3-L4', action Trust

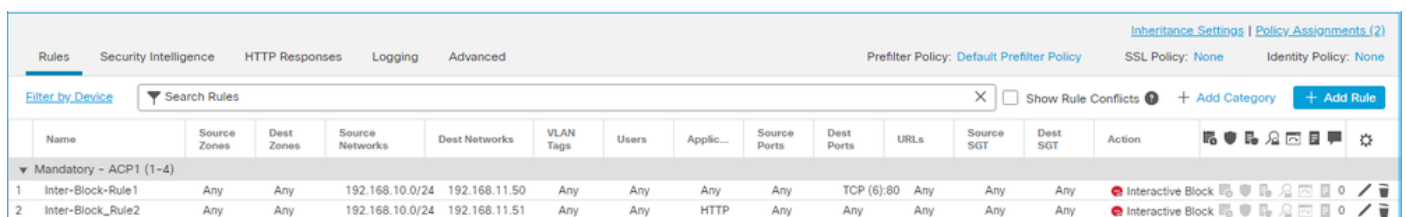
192.168.10.50-41922 > 192.168.11.50-80 6 AS 1-1 I 19 MidRecovery data sent for rule id: 268438858,rule\_

## Use Cases

Used to monitor network activity and generate a Connection Event

## ACP Interactive Block Action

An Interactive Block rule configured on FMC UI:



The screenshot shows the Cisco FMC Rules configuration page. The 'Rules' tab is selected. The page displays a table of rules. The rule 'Inter-Block\_Rule2' is highlighted, showing its configuration details. The rule is configured with the following settings:

Name	Source Zones	Dest Zones	Source Networks	Dest Networks	VLAN Tags	Users	Applic...	Source Ports	Dest Ports	URLs	Source SGT	Dest SGT	Action
1 Inter-Block-Rule1	Any	Any	192.168.10.0/24	192.168.11.50	Any	Any	Any	Any	TCP (6):80	Any	Any	Any	Interactive Block
2 Inter-Block_Rule2	Any	Any	192.168.10.0/24	192.168.11.51	Any	Any	HTTP	Any	Any	Any	Any	Any	Interactive Block

The Interactive Block rule is deployed on the FTD LINA engine as a **permit** action and to the Snort engine as a bypass rule:

```
<#root>
```

```
firepower#
```

```
show access-list
```

```
...  
access-list CSM_FW_ACL_ line 9 remark rule-id 268438864: L7 RULE: Inter-Block-Rule1
```

```
access-list CSM_FW_ACL_ line 10 advanced
```

```
permit
```

```
tcp 192.168.10.0 255.255.255.0 host 192.168.11.50 eq www rule-id 268438864 (hitcnt=3) 0xba785fc0  
access-list CSM_FW_ACL_ line 11 remark rule-id 268438865: ACCESS POLICY: ACP1 - Mandatory  
access-list CSM_FW_ACL_ line 12 remark rule-id 268438865: L7 RULE: Inter-Block_Rule2  
access-list CSM_FW_ACL_ line 13 advanced
```

```
permit
```

```
ip 192.168.10.0 255.255.255.0 host 192.168.11.51 rule-id 268438865 (hitcnt=0) 0x622350d0
```

Snort engine:

```
<#root>
```

```
admin@firepower:~$
```

```
cat /var/sf/detection_engines/9e080e5c-adc3-11ea-9d37-44884cf7e9ba/ngfw.rules
```

```
...
```

```
# Start of AC rule.  
268438864
```

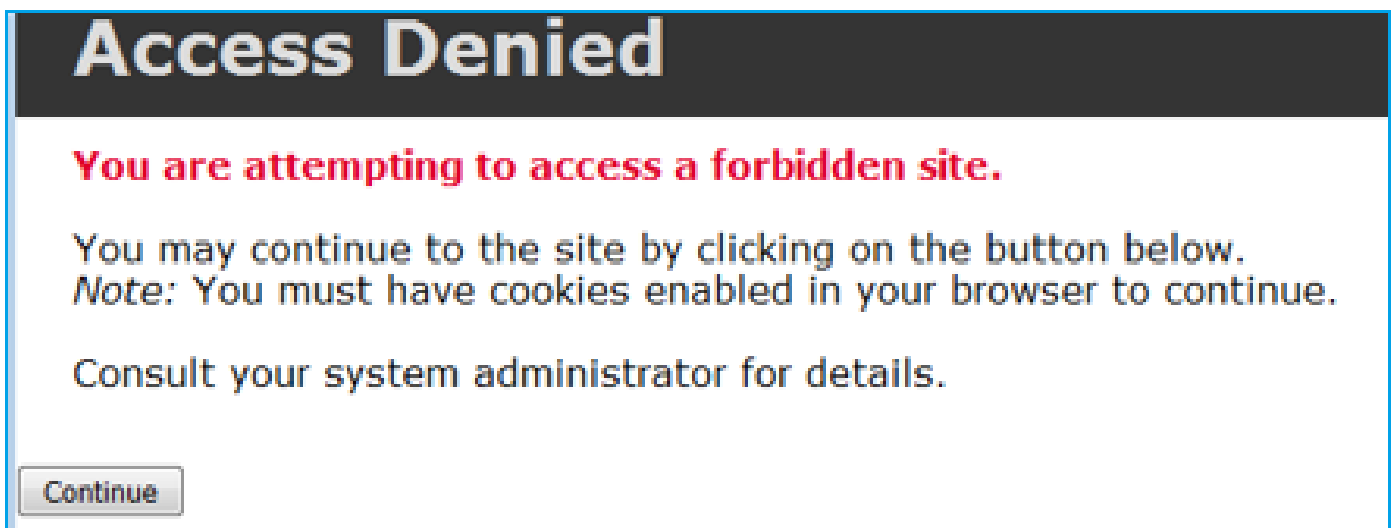
```
bypass
```

```
any 192.168.10.0 24 any any 192.168.11.50 32 80 any 6  
# End rule 268438864  
268438865
```

```
bypass
```

```
any 192.168.10.0 24 any any 192.168.11.51 32 any any any (appid 676:1) (ip_protos 6, 17)  
# End rule 268438865
```

Interactive Block Rule prompts the user that the destination is forbidden



**Access Denied**

**You are attempting to access a forbidden site.**

You may continue to the site by clicking on the button below.  
*Note:* You must have cookies enabled in your browser to continue.

Consult your system administrator for details.

Continue

By default, the firewall allows to bypass the block for 600 seconds:

Rules	Security Intelligence	HTTP Responses	Logging	Advanced
<b>General Settings</b>				
Maximum URL characters to store in connection events				1024
Allow an Interactive Block to bypass blocking for (seconds)				600
Retry URL cache miss lookup				Yes
Enable Threat Intelligence Director				Yes
Inspect traffic during policy apply				Yes

In the **system support trace** output you can see that initially the firewall blocks the traffic and shows the block page:

```
<#root>
```

```
>
```

```
system support trace
```

```
...
```

```
192.168.10.52-58717 - 192.168.11.50-80 6 AS 1-1 CID 0 Packet: TCP, ACK, seq 983273680, ack 2014879580
192.168.10.52-58717 - 192.168.11.50-80 6 AS 1-1 CID 0 AppID: service HTTP (676), application unknown (0)
192.168.10.52-58717 > 192.168.11.50-80 6 AS 1-1 I 22 Starting AC with minimum 2, 'Inter-Block-Rule1', a
192.168.10.52-58717 > 192.168.11.50-80 6 AS 1-1 I 22 match rule order 2, 'Inter-Block-Rule1',
```

```
action Interactive
```

```
192.168.10.52-58717 > 192.168.11.50-80 6 AS 1-1 I 22
```

```
bypass action sending HTTP interactive response of 1093 bytes
```

```
192.168.10.52-58717 - 192.168.11.50-80 6 AS 1-1 CID 0 Firewall: interactive block rule, 'Inter-Block-Ru
192.168.10.52-58717 - 192.168.11.50-80 6 AS 1-1 CID 0 Snort: processed decoder alerts or actions queue,
192.168.10.52-58717 > 192.168.11.50-80 6 AS 1-1 I 22 deleting firewall session flags = 0x800, fwFlags =
192.168.10.52-58717 - 192.168.11.50-80 6 AS 1-1 CID 0 Snort id 22, NAP id 1, IPS id 0,
```

```
Verdict BLACKLIST
```

```
192.168.10.52-58717 - 192.168.11.50-80 6 AS 1-1 CID 0 ==>
```

```
Blocked by Firewall
```

```
Verdict reason is sent to DAQ
```

Once the user selects **Continue** (or refreshes the browser page) the debug shows that the packets are allowed by the same rule which mimics and **Allow** action:

```
<#root>
```

```
192.168.10.52-58718 - 192.168.11.50-80 6 AS 1-1 CID 0 Packet: TCP, ACK, seq 1357413630, ack 2607625293
192.168.10.52-58718 - 192.168.11.50-80 6 AS 1-1 CID 0 AppID: service HTTP (676), application unknown (0)
192.168.10.52-58718 > 192.168.11.50-80 6 AS 1-1 I 8 Starting AC with minimum 2, 'Inter-Block-Rule1', an
```

```

192.168.10.52-58718 > 192.168.11.50-80 6 AS 1-1 I 8 match rule order 2, 'Inter-Block-Rule1',
action Interactive

192.168.10.52-58718 > 192.168.11.50-80 6 AS 1-1 I 8
bypass action interactive bypass

192.168.10.52-58718 > 192.168.11.50-80 6 AS 1-1 I 8
allow action

192.168.10.52-58718 - 192.168.11.50-80 6 AS 1-1 CID 0 Firewall: allow rule, 'Inter-Block-Rule1', allow
192.168.10.52-58718 - 192.168.11.50-80 6 AS 1-1 CID 0 Firewall: interactive bypass rule, 'Inter-Block-R
192.168.10.52-58718 - 192.168.11.50-80 6 AS 1-1 CID 0 Snort id 8, NAP id 1, IPS id 0,

Verdict PASS

```

## Use Cases

Show a warning page to web users and give them the option to continue.

## ACP Interactive Block with Reset Action

An Interactive Block with reset rule configured on FMC UI:

Name	Source Zones	Destination Zones	Source Networks	Destination Networks	VLAN Tags	Users	Applications	Source Ports	Destination Ports	URLs	Source SGT	Destination SGT	Action
1 Inter-Block-Rule1	Any	Any	192.168.10.0/24	192.168.11.50	Any	Any	Any	Any	TCP (6):80	Any	Any	Any	Interactive Block with reset
2 Inter-Block_Rule2	Any	Any	192.168.10.0/24	192.168.11.51	Any	Any	HTTP	Any	Any	Any	Any	Any	Interactive Block with reset

The Interactive Block with reset rule is deployed on FTD LINA engine as a **permit** action and to Snort engine as intreset rule:

```

<#root>

firepower#

show access-list

...
access-list CSM_FW_ACL_ line 9 remark rule-id 268438864: L7 RULE: Inter-Block-Rule1
access-list CSM_FW_ACL_ line 10 advanced

permit

tcp 192.168.10.0 255.255.255.0 host 192.168.11.50 eq www rule-id 268438864 (hitcnt=13) 0xba785fc0
access-list CSM_FW_ACL_ line 11 remark rule-id 268438865: ACCESS POLICY: ACP1 - Mandatory
access-list CSM_FW_ACL_ line 12 remark rule-id 268438865: L7 RULE: Inter-Block_Rule2
access-list CSM_FW_ACL_ line 13 advanced

permit

ip 192.168.10.0 255.255.255.0 host 192.168.11.51 rule-id 268438865 (hitcnt=0) 0x622350d0

```

Snort engine:

```
<#root>

# Start of AC rule.
268438864

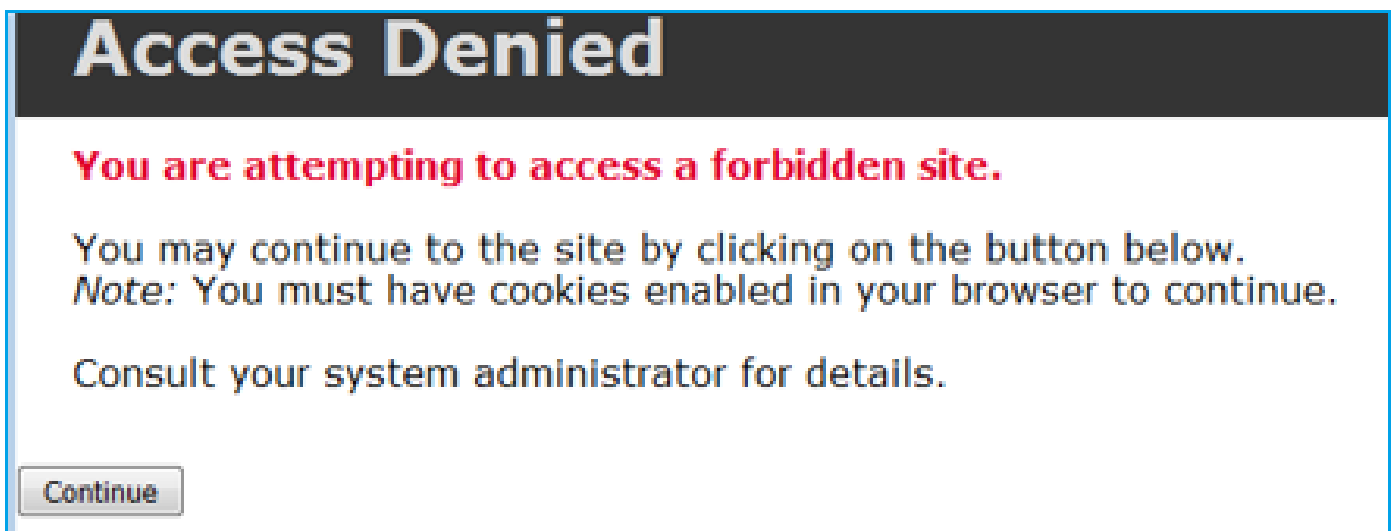
intreset

any 192.168.10.0 24 any any 192.168.11.50 32 80 any 6
# End rule 268438864
268438865

intreset

any 192.168.10.0 24 any any 192.168.11.51 32 any any any (appid 676:1) (ip_protos 6, 17)
# End rule 268438865
```

Like the Block with Reset, the user can select the **Continue** option:



In the Snort debug the action shown in Interactive Reset:

```
<#root>
>
system support trace

Enable firewall-engine-debug too? [n]:
y

Please specify an IP protocol: tcp
Please specify a client IP address:
192.168.10.52

Please specify a client port:
Please specify a server IP address:
192.168.11.50

Please specify a server port:
Monitoring packet tracer and firewall debug messages
```

```

192.168.10.52-58958 - 192.168.11.50-80 6 AS 1-1 CID 0 Packet: TCP, SYN, seq 3232128039
192.168.10.52-58958 - 192.168.11.50-80 6 AS 1-1 CID 0 Session: new snort session
192.168.10.52-58958 - 192.168.11.50-80 6 AS 1-1 CID 0 AppID: service unknown (0), application unknown (0)
192.168.10.52-58958 > 192.168.11.50-80 6 AS 1-1 I 24 new firewall session
192.168.10.52-58958 > 192.168.11.50-80 6 AS 1-1 I 24 Starting AC with minimum 2, 'Inter-Block-Rule1', a
192.168.10.52-58958 > 192.168.11.50-80 6 AS 1-1 I 24 match rule order 2, 'Inter-Block-Rule1', action In
192.168.10.52-58958 > 192.168.11.50-80 6 AS 1-1 I 24 MidRecovery data sent for rule id: 268438864,rule
192.168.10.52-58958 > 192.168.11.50-80 6 AS 1-1 I 24 HitCount data sent for rule id: 268438864,
192.168.10.52-58958 - 192.168.11.50-80 6 AS 1-1 CID 0 Snort id 24, NAP id 1, IPS id 0, Verdict PASS

192.168.11.50-80 - 192.168.10.52-58958 6 AS 1-1 CID 0 Packet: TCP, SYN, ACK, seq 2228213518, ack 323212
192.168.11.50-80 - 192.168.10.52-58958 6 AS 1-1 CID 0 AppID: service unknown (0), application unknown (0)
192.168.11.50-80 - 192.168.10.52-58958 6 AS 1-1 CID 0 Snort id 24, NAP id 1, IPS id 0, Verdict PASS

192.168.10.52-58958 - 192.168.11.50-80 6 AS 1-1 CID 0 Packet: TCP, ACK, seq 3232128040, ack 2228213519
192.168.10.52-58958 - 192.168.11.50-80 6 AS 1-1 CID 0 AppID: service unknown (0), application unknown (0)
192.168.10.52-58958 - 192.168.11.50-80 6 AS 1-1 CID 0 Snort id 24, NAP id 1, IPS id 0, Verdict PASS

192.168.10.52-58958 - 192.168.11.50-80 6 AS 1-1 CID 0 Packet: TCP, ACK, seq 3232128040, ack 2228213519
192.168.10.52-58958 - 192.168.11.50-80 6 AS 1-1 CID 0 AppID: service HTTP (676), application unknown (0)
192.168.10.52-58958 > 192.168.11.50-80 6 AS 1-1 I 24 Starting AC with minimum 2, 'Inter-Block-Rule1', a
192.168.10.52-58958 > 192.168.11.50-80 6 AS 1-1 I 24 match rule order 2, 'Inter-Block-Rule1',

action Interactive Reset

192.168.10.52-58958 > 192.168.11.50-80 6 AS 1-1 I 24

bypass action sending HTTP interactive response of 1093 bytes

192.168.10.52-58958 - 192.168.11.50-80 6 AS 1-1 CID 0 Firewall: interactive block rule, 'Inter-Block-Ru
192.168.10.52-58958 - 192.168.11.50-80 6 AS 1-1 CID 0 Snort: processed decoder alerts or actions queue,
192.168.10.52-58958 > 192.168.11.50-80 6 AS 1-1 I 24 deleting firewall session flags = 0x800, fwFlags =
192.168.10.52-58958 - 192.168.11.50-80 6 AS 1-1 CID 0 Snort id 24, NAP id 1, IPS id 0,

Verdict BLACKLIST

192.168.10.52-58958 - 192.168.11.50-80 6 AS 1-1 CID 0 ==>>>

Blocked by Firewall

Verdict reason is sent to DAQ

```

At this point, the block page is shown to the end-user. If the user selects **Continue** (or refreshes the web page) the same rule matches which this time allows the traffic through:

<#root>

```

192.168.10.52-58962 - 192.168.11.50-80 6 AS 1-1 CID 0 Packet: TCP, ACK, seq 1593478294, ack 3135589307
192.168.10.52-58962 - 192.168.11.50-80 6 AS 1-1 CID 0 AppID: service HTTP (676), application unknown (0)
192.168.10.52-58962 > 192.168.11.50-80 6 AS 1-1 I 14 Starting AC with minimum 2, 'Inter-Block-Rule1', a
192.168.10.52-58962 > 192.168.11.50-80 6 AS 1-1 I 14 match rule order 2, 'Inter-Block-Rule1',

action Interactive Reset

192.168.10.52-58962 > 192.168.11.50-80 6 AS 1-1 I 14

bypass action interactive bypass

192.168.10.52-58962 > 192.168.11.50-80 6 AS 1-1 I 14 allow action
192.168.10.52-58962 - 192.168.11.50-80 6 AS 1-1 CID 0 Firewall: allow rule, 'Inter-Block-Rule1', allow
192.168.10.52-58962 - 192.168.11.50-80 6 AS 1-1 CID 0 Firewall: interactive bypass rule, 'Inter-Block-Ru

```



```
192.168.10.52-58962 - 192.168.11.50-80 6 AS 1-1 CID 0 Snort id 14, NAP id 1, IPS id 0,
```

```
Verdict PASS
```

```
192.168.11.50-80 - 192.168.10.52-58962 6 AS 1-1 CID 0 Packet: TCP, ACK, seq 3135589307, ack 1593478786
192.168.11.50-80 - 192.168.10.52-58962 6 AS 1-1 CID 0 AppID: service HTTP (676), application unknown (0)
192.168.11.50-80 - 192.168.10.52-58962 6 AS 1-1 CID 0 Firewall: allow rule, 'Inter-Block-Rule1', allow
192.168.11.50-80 - 192.168.10.52-58962 6 AS 1-1 CID 0 Firewall: interactive bypass rule, 'Inter-Block-R
192.168.11.50-80 - 192.168.10.52-58962 6 AS 1-1 CID 0 Snort id 14, NAP id 1, IPS id 0,
```

```
Verdict PASS
```

The interactive block with reset rule sends a TCP RST to non-web traffic:

```
<#root>
```

```
firepower#
```

```
show cap CAPI | i 11.50
```

```
2: 22:13:33.112954      802.1Q vlan#202 P0 192.168.10.50.40010 > 192.168.11.50.21: S 3109534920:310
3: 22:13:33.113626      802.1Q vlan#202 P0 192.168.11.50.21 > 192.168.10.50.40010: S 3422362500:342
4: 22:13:33.113824      802.1Q vlan#202 P0 192.168.10.50.40010 > 192.168.11.50.21: . ack 3422362501
5: 22:13:33.114953      802.1Q vlan#202 P0 192.168.11.50.21 > 192.168.10.50.40010: P 3422362501:342
6: 22:13:33.114984      802.1Q vlan#202 P0 192.168.11.50.21 > 192.168.10.50.40010: P 3422362543:342
7: 22:13:33.114984      802.1Q vlan#202 P0 192.168.11.50.21 > 192.168.10.50.40010: P 3422362549:342
8: 22:13:33.115182      802.1Q vlan#202 P0 192.168.10.50.40010 > 192.168.11.50.21: . ack 3422362543
9: 22:13:33.115411      802.1Q vlan#202 P0 192.168.10.50.40010 > 192.168.11.50.21: . ack 3422362549
10: 22:13:33.115426      802.1Q vlan#202 P0 192.168.10.50.40010 > 192.168.11.50.21: . ack 3422362570
12: 22:13:34.803699      802.1Q vlan#202 P0 192.168.10.50.40010 > 192.168.11.50.21: P 3109534921:310
13: 22:13:34.804523      802.1Q vlan#202 P0 192.168.11.50.21 > 192.168.10.50.40010:
```

```
R
```

```
3422362570:3422362570(0) ack 3109534931 win 0
```

## FTD Secondary Connections and Pinholes

In older releases (for example 6.2.2, 6.2.3, etc) the Snort engine does not open pinholes for secondary connections (for example, FTD Data) if you use the **Trust** action. In recent releases, this behavior is changed and the Snort engine opens pinholes even with the **Trust** action.

## FTD Rule Guidelines

- Use Prefilter Policy Fastpath rules for big fat flows and in order to decrease latency through the box
- Use Prefilter Block rules for traffic that must be blocked based on L3/L4 conditions
- Use ACP Trust rules if you want to bypass many of the Snort checks, but still take advantage of features like Identity Policy, QoS, SI, Application detection, URL filter
- Place rules that affect less the firewall performance at the top of the Access Control Policy with the use of these guidelines:

1. Block rules (layers 1-4) - Prefilter Block
2. Allow rules (layers 1-4) - Prefilter Fastpath
3. ACP Block rules (layers 1-4)

4. Trust rules (layers 1-4)
  5. Block rules (layers 5-7 - application detection, URL filtering)
  6. Allow rules (layers 1-7 - application detection, URL filtering, Intrusion Policy/File Policy)
  7. Block rule (Default rule)
- Avoid excessive logging (log at the start or at the end and avoid both at the same time)
  - Be aware of the rule expansion, to check the number of rules in LINA

<#root>

firepower#

show access-list | include elements

access-list CSM\_FW\_ACL\_;

7 elements

; name hash: 0x4a69e3f3


## Summary

### Prefilter Actions

Rule Action (FMC UI)	LINA Action	Snort Action	Notes
Fastpath	Trust	Fastpath	Static Flow Offload to SmartNIC (4100/9300). <b>No packets</b> are sent to Snort engine.
Analyze	Permit	-	The ACP rules are checked. <b>Few or all packets</b> are sent to Snort engine for inspection. Traffic is allowed or dropped based on Snort engine verdict
Block (Prefilter)	Deny	-	Early drop by FTD LINA <b>No packets</b> are sent to Snort engine

### ACP Actions

Rule Action (FMC UI)	Additional Conditions	LINA Action	Snort Action	Notes
Block	The rule matches L3/L4 conditions	Deny	Deny	
Block	The rule has L7 conditions	Permit	Deny	
Allow		Permit	Allow	6.3+ supports Dynamic Flow Offload (4100/9300)
Trust	(SI, QoS, or ID) enabled	Permit	Fastpath	6.3+ supports Dynamic Flow Offload (4100/9300)
Trust	(SI, QoS, and ID) disabled	Trust	Fastpath	Static Flow Offload (4100/9300)
Monitor		Permit	Audit	Monitor Rule doesn't drop or permit traffic, but it generates a Connection Event. The packet is checked against subsequent rules and it is either allowed or dropped
Block with reset		Permit	Reset	When a packet matches Block with reset rule FTD sends a TCP Reset packet or an ICMP Type 3 Code 13 Destination Unreachable (Administratively filtered) message
Interactive Block		Permit	Bypass	Interactive Block Rule prompts the user that the destination is forbidden. If bypassed, by default, the firewall allows to bypass the block for 600 seconds
Interactive Block with reset		Permit	Intreset	Same as Interactive Block with the addition of a TCP RST in case of non-web traffic

 **Note:** As from 6.3 FTD software code Dynamic flow offload can offload connections that meet additional criteria; for example, trusted packets that require Snort inspection. Check the **Offload Large Connections (Flows)** section from the **Firepower Management Center Configuration Guide** for more details

## Related Information

- [FTD Access Control Rules](#)
- [FTD Prefiltering and Prefilter Policies](#)
- [Analyze Firepower Firewall Captures to Effectively Troubleshoot Network Issues](#)
- [Working with Firepower Threat Defense \(FTD\) Captures and Packet-Tracer](#)
- [Configure Logging on FTD via FMC](#)
- [Technical Support & Documentation - Cisco Systems](#)
- [Offload Large Connections](#)