Interpret Firepower Threat Defense TCP Connection Flags (Connection Build-Up and Teardown)

Contents

Introduction

This document describes how to troubleshoot TCP connections through the Firepower Threat Defense (FTD).

Prerequisites

Requirements

Cisco recommends that you have knowledge of these topics:

- Basic knowledge of the TCP Communication Protocol.
- Basic knowledge of the FTD CLI.

Components Used

This document is not restricted to specific software and hardware versions.

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.

Troubleshoot TCP Connections

When you troubleshoot TCP connections through the FTD, the connection flags shown for each connection provides a wealth of information about the state of TCP connections through the FTD. This information can be used to troubleshoot problems with the FTD, as well as problems elsewhere in the network.

```
Disclaimer: The information in this document was created based on FTD devices on version 7.0 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.

Since all FTD interfaces have a Security Level of 0, the interface order in the show conn output is based on the interface number. Specifically, the interface with a higher Virtual Platform Interface Number (VPIF) is shown first.

```
Disclaimer : The show conn output can be too long, hence it is recommended to use 'terminal
pager' or write into a file saved in disk0: such as 'show conn | redirect filename.txt'

firepower# show conn
3 in use, 22 most used
Inspect Snort:
preserve-connection: 3 enabled, 0 in effect, 22 most enabled, 0 most in effect

TCP ISP2 192.168.50.14:35518 Inside 192.168.45.130:22, idle 0:10:00, bytes 7164, flags UIO N1
TCP ISP2 192.168.50.14:80 Inside 192.168.45.130:54554, idle 0:00:13, bytes 0, flags U N1
TCP Inside 192.168.45.130:34070 ISP1 10.31.104.78:3128, idle 0:00:02, bytes 1187822, flags UIO
N1
```

You can see the interface VPIF value from the output of show interface detailcommand.

```
firepower# show interface detail | i Interface number is|Interface
Interface GigabitEthernet0/0 "ISP1", is up, line protocol is up
Control Point Interface States:
      Interface number is 3
Interface config status is active
Interface state is active
Interface GigabitEthernet0/1 "Inside", is up, line protocol is up
Control Point Interface States:
      Interface number is 4
Interface config status is active
Interface state is active
Interface GigabitEthernet0/2 "DMZ", is up, line protocol is up
Control Point Interface States:
      Interface number is 5
Interface config status is active
Interface state is active
Interface GigabitEthernet0/3 "ISP2", is up, line protocol is up
Control Point Interface States:
      Interface number is 6
Interface config status is active
Interface state is active
```

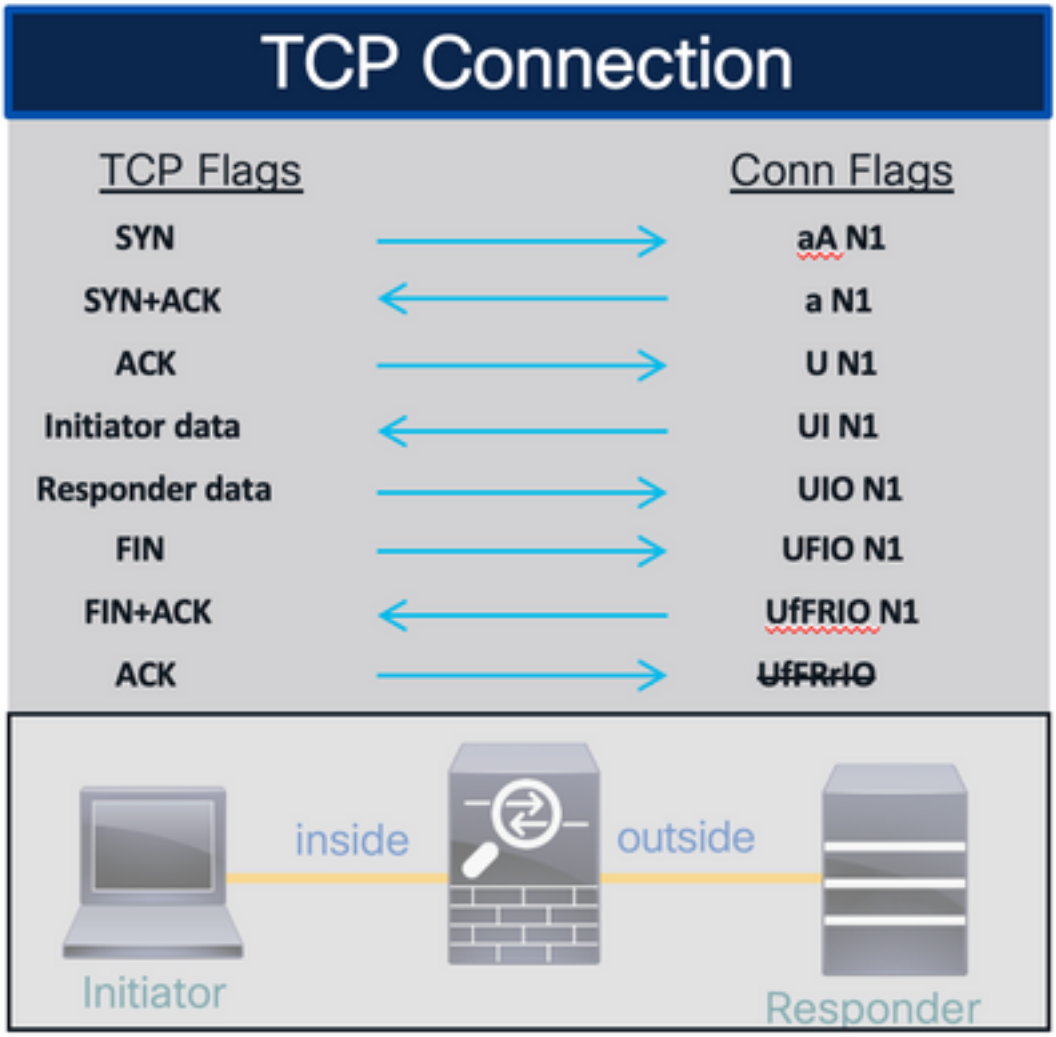
The show conn longand show conn detailcommands provide details about the Initiator and the Responder of the connection.

```
firepower# show conn long
3 in use, 22 most used
Inspect Snort:
preserve-connection: 3 enabled, 0 in effect, 22 most enabled, 0 most in effect
Flags: A - awaiting responder ACK to SYN, a - awaiting initiator ACK to SYN,
B - TCP probe for server certificate,
b - TCP state-bypass or nailed,
C - CTIQBE media, c - cluster centralized,
D - DNS, d - dump, E - outside back connection, e - semi-distributed,
F - initiator FIN, f - responder FIN,
```

```
G - group, g - MGCP, H - H.323, h - H.225.0, I - initiator data,
i - incomplete, J - GTP, j - GTP data, K - GTP t3-response
k - Skinny media, L - decap tunnel, M - SMTP data, m - SIP media
N - inspected by Snort (1 - preserve-connection enabled, 2 - preserve-connection in effect)
n - GUP, O - responder data, o - offloaded,
P - inside back connection, p - passenger flow
q - SQL*Net data, R - initiator acknowledged FIN,
R - UDP SUNRPC, r - responder acknowledged FIN,
T - SIP, t - SIP transient, U - up,
V - VPN orphan, v - M3UA W - WAAS,
w - secondary domain backup,
X - inspected by service module,
x - per session, Y - director stub flow, y - backup stub flow,
Z - Scansafe redirection, z - forwarding stub flow

TCP ISP2: 192.168.50.14/35518 (192.168.50.14/35518) Inside: 192.168.45.130/22
(192.168.45.130/22), flags UIO N1, idle 9m13s, uptime 9m17s, timeout 1h0m, bytes 7164
Initiator: 192.168.50.14, Responder: 192.168.45.130
Connection lookup keyid: 168317598

TCP ISP2: 192.168.50.14/80 (192.168.50.14/80) Inside: 192.168.45.130/54554
(192.168.45.130/54554), flags U N1, idle 0s, uptime 10s, timeout 1h0m, bytes 0
Initiator: 192.168.45.130, Responder: 192.168.50.14
Connection lookup keyid: 168367034

TCP Inside: 192.168.45.130/34070 (192.168.45.130/34070) ISP1: 10.31.104.78/3128
(10.31.104.78/3128), flags UIO N1, idle 0s, uptime 46s, timeout 1h0m, bytes 617331
Initiator: 192.168.45.130, Responder: 10.31.104.78
Connection lookup keyid: 168227654
```

FTD TCP Connection Flags

This table shows the FTD TCP Connection flags at different stages of the TCP state machine. In FTD, the connection flags are the same for inbound and outbound connections since the security-levels are always '0'. These flags can be seen with the **show conn** command on the FTD.



TCP Connection Flag Values

This table shows the TCP Connection Flags that are removed and added upon the receipt of a packet.

Flags **REMOVED** upon Receipt of Packet	a	Awaiting Initiator ACK to SYN
	A	Awaiting Responder ACK to SYN
Flags **ADDED** upon Receipt of Packet	U	Up - 3-way Handshake Complete
	I	Received Initiator Data
	O	Received Responder Data
	F	Received Initiator FIN
	f	Received Responder FIN
	R	Received Initiator ACK to FIN
	N1	Inspected by Snort with preserve-connection enabled
	N2	Inspected by Snort with preserve-connection in effect

To view all of the possible flags in a connection use the **show conn detail** command.

```
firepower# show conn detail
1 in use, 22 most used
Inspect Snort:
preserve-connection: 1 enabled, 0 in effect, 22 most enabled, 0 most in effect
Flags: A - awaiting responder ACK to SYN, a - awaiting initiator ACK to SYN,
B - TCP probe for server certificate,
b - TCP state-bypass or nailed,
C - CTIQBE media, c - cluster centralized,
D - DNS, d - dump, E - outside back connection, e - semi-distributed,
F - initiator FIN, f - responder FIN,
G - group, g - MGCP, H - H.323, h - H.225.0, I - initiator data,
i - incomplete, J - GTP, j - GTP data, K - GTP t3-response
k - Skinny media, L - decap tunnel, M - SMTP data, m - SIP media
N - inspected by Snort (1 - preserve-connection enabled, 2 - preserve-connection in effect)
n - GUP, O - responder data, o - offloaded,
P - inside back connection, p - passenger flow
q - SQL*Net data, R - initiator acknowledged FIN,
R - UDP SUNRPC, r - responder acknowledged FIN,
T - SIP, t - SIP transient, U - up,
V - VPN orphan, v - M3UA W - WAAS,
w - secondary domain backup,
X - inspected by service module,
x - per session, Y - director stub flow, y - backup stub flow,
Z - Scansafe redirection, z - forwarding stub flow
```