

Vastleggingen van de Firepower-firewall analyseren om netwerkproblemen effectief te troubleshooten

Inhoud

[Inleiding](#)

[Voorwaarden](#)

[Vereisten](#)

[Gebruikte componenten](#)

[Achtergrondinformatie](#)

[Hoe verzamelt en exporteert u opnamen over de NGFW-productfamilie?](#)

[FXOS-opnamen verzamelen](#)

[FTD Lina Captures inschakelen en verzamelen](#)

[FTD-snortopnamen inschakelen en verzamelen](#)

[Problemen oplossen](#)

[Situatie 1. Geen TCP/SYN op uitgaande interface](#)

[Capture Analysis](#)

[Aanbevolen acties](#)

[Mogelijke oorzaken en aanbevolen acties Samenvatting](#)

[Situatie 2. TCP/SYN vanaf client, TCP/RST vanaf server](#)

[Capture Analysis](#)

[Aanbevolen acties](#)

[Situatie 3. TCP/drieweg-handdruk + RST van één eindpunt](#)

[Capture Analysis](#)

[3.1 - TCP 3-weg handdruk + vertraagde RST van de client](#)

[Aanbevolen acties](#)

[3.2 - TCP 3-weg handdruk + vertraagde FIN/ACK van client + vertraagde RST van de server](#)

[Aanbevolen acties](#)

[3.3 - TCP 3-weg handdruk + vertraagde RST van de client](#)

[Aanbevolen acties](#)

[3.4 - TCP 3-weg handdruk + directe RST van de server](#)

[Aanbevolen acties](#)

[Situatie 4. TCP/RST vanaf de client](#)

[Capture Analysis](#)

[Aanbevolen acties](#)

[Situatie 5. Langzame TCP-overdracht \(scenario 1\)](#)

[Scenario 1. Langzame overdracht](#)

[Capture Analysis](#)

[Aanbevolen acties](#)

[Scenario 2. Snelle overdracht](#)

[Situatie 6. Langzame TCP-overdracht \(scenario 2\)](#)

[Capture Analysis](#)

[Aanbevolen acties](#)

[Situatie 7. Probleem met TCP-connectiviteit \(pakketcorruptie\)](#)

[Capture Analysis](#)

[Aanbevolen acties](#)

[Situatie 8. UDP-connectiviteitsprobleem \(ontbrekende pakketten\)](#)

[Capture Analysis](#)

[Aanbevolen acties](#)

[Situatie 9. Connectiviteitsprobleem met HTTPS \(scenario 1\)](#)

[Capture Analysis](#)

[Aanbevolen acties](#)

[Situatie 10. Connectiviteitsprobleem met HTTPS \(scenario 2\)](#)

[Capture Analysis](#)

[Aanbevolen acties](#)

[Situatie 11. IPv6-connectiviteitsprobleem](#)

[Capture Analysis](#)

[Aanbevolen acties](#)

[Situatie 12. Probleem met intermitterende connectiviteit \(ARP-vergiftiging\)](#)

[Capture Analysis](#)

[Aanbevolen acties](#)

[Situatie 13. Identificeer SNMP-objectidentificatiecodes \(OID's\) die CPU-fouten veroorzaken](#)

[Capture Analysis](#)

[Aanbevolen acties](#)

[Gerelateerde informatie](#)

Inleiding

In dit document worden verschillende technieken voor analyse van pakketvastlegging omschreven die bedoeld zijn om netwerkproblemen effectief op te lossen.

Voorwaarden

Vereisten

Cisco raadt kennis van de volgende onderwerpen aan:

- Firepower platform architectuur
- NGFW-logs
- NGFW-pakkettracer

Bovendien, alvorens u begint pakketopnamen te analyseren is het hoogst raadzaam om aan deze vereisten te voldoen:

- **Ken de protocolhandeling** - begin geen pakketopname te controleren als u niet begrijpt hoe het opgenomen protocol werkt.
- **Ken de topologie** - U moet de vervoerapparaten van begin tot eind kennen. Als dit niet mogelijk is, moet u tenminste weten de stroomopwaarts en stroomafwaarts apparaten.
- **Ken het apparaat** - U moet weten hoe uw apparaat pakketten verwerkt, wat de betrokken interfaces zijn (ingangen/uitgangen), wat de architectuur van het apparaat is en wat de verschillende opnamepunten zijn.
- **Weet de configuratie** - U moet weten hoe een pakketstroom door het apparaat moet worden verwerkt in termen van:
 - Routing/uitgaande interface
 - Toegepast beleid
 - Netwerkadresomzetting (NAT)
- **Ken de beschikbare gereedschappen** - Samen met de opnamen, is het aan te raden om klaar te zijn om andere gereedschappen en technieken toe te passen (zoals houtkap en tracers) en deze indien nodig te correleren met de opgenomen pakketten

Gebruikte componenten

De informatie in dit document is gebaseerd op de volgende software- en hardware-versies:

- De meeste scenario's zijn gebaseerd op FP4140 lopende FTD-software 6.5.x.
- FMC hardloopsoftware 6.5.x.

De informatie in dit document is gebaseerd op de apparaten in een specifieke laboratoriumomgeving. Alle apparaten die in dit document worden beschreven, hadden een opgeschoonde (standaard)configuratie. Als uw netwerk live is, moet u zorgen dat u de potentiële impact van elke opdracht begrijpt.

Achtergrondinformatie

Packet Capture is een van de meest over het hoofd geziene tools voor probleemoplossing die momenteel beschikbaar zijn. Cisco TAC lost dagelijks veel problemen op met de analyse van opgenomen gegevens.

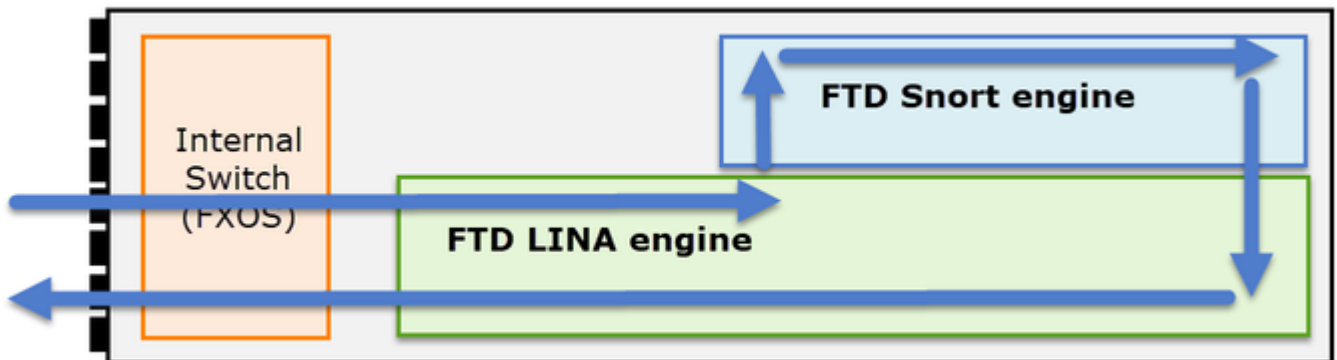
Het doel van dit document is netwerk- en security engineers te helpen gemeenschappelijke netwerkproblemen te identificeren en op te lossen die voornamelijk gebaseerd zijn op pakketopnameanalyse.

Alle scenario's die in dit document worden gepresenteerd, zijn gebaseerd op echte gebruikerscases die in het Cisco Technical Assistance Center (TAC) worden gezien.

Het document behandelt het pakket en neemt het op vanuit een Cisco Next-generation firewall (NGFW) oogpunt, maar dezelfde concepten zijn ook van toepassing op andere apparaattypen.

Hoe verzamelt en exporteert u opnamen over de NGFW-productfamilie?

In het geval van een FirePOWER-applicatie (1xxx, 21xx, 41xx, 93xx) en een FirePOWER Threat Defence (FTD) kan een pakketverwerking worden gevisualiseerd zoals in de afbeelding.



1. Een pakket gaat de toegangsinterface in en het wordt behandeld door de switch van de chassisbinnenkant.
2. Het pakket gaat de FTD Lina engine in die voornamelijk L3/L4-controles uitvoert.
3. Als het beleid vereist dat het pakket wordt geïnspecteerd door de Snort engine (voornamelijk L7 inspection).
4. De snort engine geeft een oordeel voor het pakket terug.
5. De LINA-engine wijst het pakket af of stuurt het door op basis van het Snort-oordeel.
6. Het pakket gaat met de switch van het chassis naar binnen.

Op basis van de getoonde architectuur kunnen de FTD-opnamen op drie (3) verschillende plaatsen worden gemaakt:

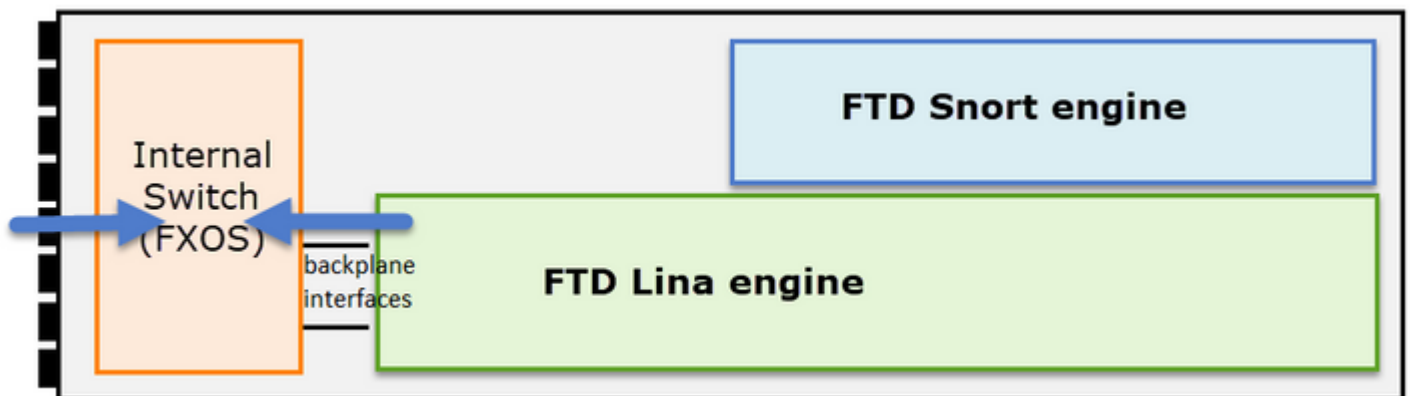
- FXOS
- FTD Lina-motor
- FTD Snort-engine

FXOS-opnamen verzamelen

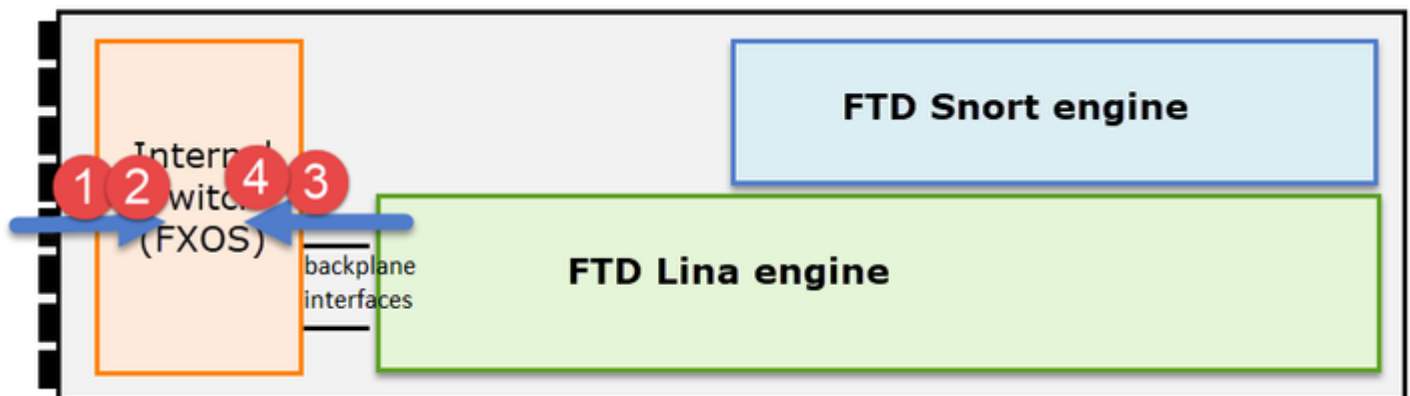
Het proces wordt in dit document beschreven:

https://www.cisco.com/c/en/us/td/docs/security/firepower/fxos/fxos271/web-guide/b_GUI_FXOS_ConfigGuide_271/troubleshooting.html#concept_E8823CC63C934A909BBC0DF12F301DE1

FXOS-opnamen kunnen alleen vanuit het oogpunt van de inwendige switch in de toegangsrichting worden genomen; zie de afbeelding hier.



Hier getoond, zijn dit twee opnamepunten per richting (wegens binnenarchitectuur van de switch).



Opgenomen pakketten in punten 2, 3 en 4 hebben een virtuele netwerktag (VNTag).

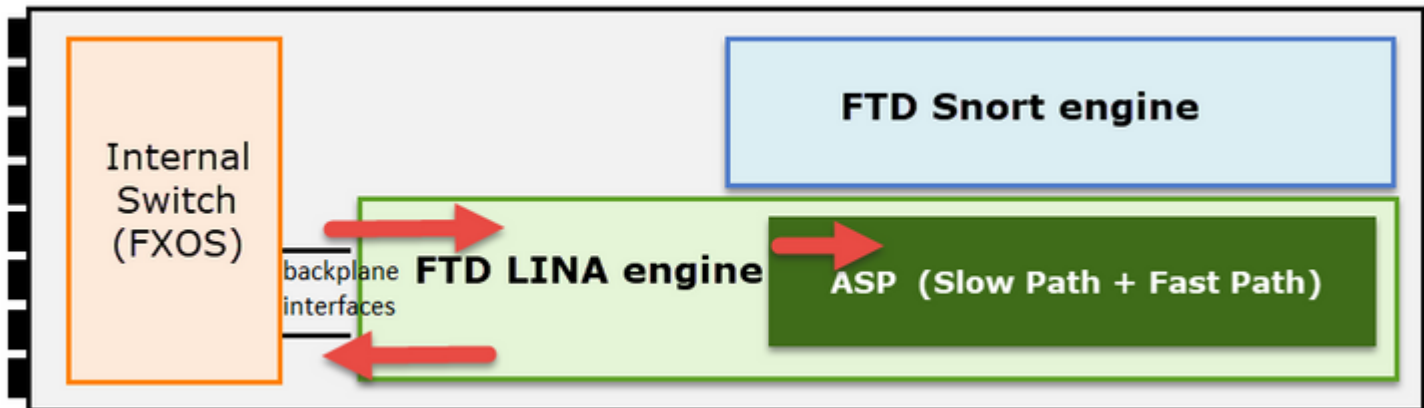
Opmerking: FXOS-opnamen op chassisniveau zijn alleen beschikbaar op FP41xx- en FP93xx-platforms. FP1xxx en FP21xx bieden deze mogelijkheid niet.

FTD Lina Captures inschakelen en verzamelen

Hoofdopnamepunten:

- Ingress-interface
- Uitgaande interface

- Sneler security pad (ASP)



U kunt Firepower Management Center User Interface (FMC UI) of FTD CLI gebruiken om de FTD Lina-opnamen in te schakelen en te verzamelen.

Opname vanaf CLI op de INSIDE-interface inschakelen:

```
<#root>
firepower#
capture CAPI interface INSIDE match icmp host 192.168.103.1 host 192.168.101.1
```

Deze opname matcht het verkeer tussen IP's 192.168.103.1 en 192.168.101.1 in beide richtingen.

Schakel ASP Capture in om alle pakketten te zien die door de FTD Lina-engine zijn gedropt:

```
<#root>
firepower#
capture ASP type asp-drop all
```

Exporteren van een FTD-lijnopname naar een FTP-server:

```
<#root>
firepower#
copy /pcap capture:CAPI ftp://ftp_username:ftp_password@192.168.78.73/CAPI.pcap
```

Exporteer een FTD Lina-opname naar een TFTP-server:

```
<#root>
firepower#
```

```
copy /pcap capture:CAPI tftp://192.168.78.73
```

Vanaf FMC 6.2.x versie kunt u FTD Lina opnamen van FMC UI inschakelen en verzamelen.

Een andere manier om FTD-opnamen te verzamelen van een door FMC beheerde firewall is dit.

Stap 1

In het geval van LINA of ASP Capture kopieert u de opname naar de FTD-schijf.

```
<#root>
firepower#
copy /pcap capture:capin disk0:capin.pcap
```

```
Source capture name [capin]?
```

```
Destination filename [capin.pcap]?
!!!!
```

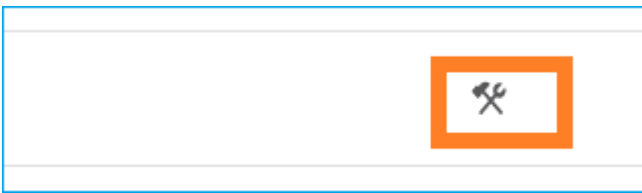
Stap 2

Navigeer naar expert-modus, lokaliseer de opgeslagen opname en kopieer deze naar de /ngfw/var/common-locatie:

```
<#root>
firepower#
Console connection detached.
>
expert
admin@firepower:~$
sudo su
Password:
root@firepower:/home/admin#
cd /mnt/disk0
root@firepower:/mnt/disk0#
ls -al | grep pcap
-rwxr-xr-x 1 root root    24 Apr 26 18:19 CAPI.pcap
-rwxr-xr-x 1 root root 30110 Apr  8 14:10
capin.pcap
-rwxr-xr-x 1 root root  6123 Apr  8 14:11 capin2.pcap
root@firepower:/mnt/disk0#
cp capin.pcap /ngfw/var/common
```

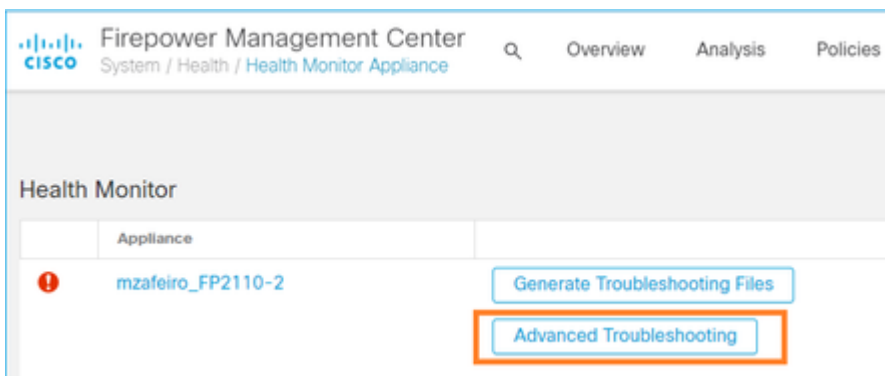
Stap 3

Meld u aan bij het VCC dat de FTD beheert en navigeer naar **Apparaten > Apparaatbeheer**. Zoek het FTD-apparaat en selecteer het pictogram **Probleemoplossing**:

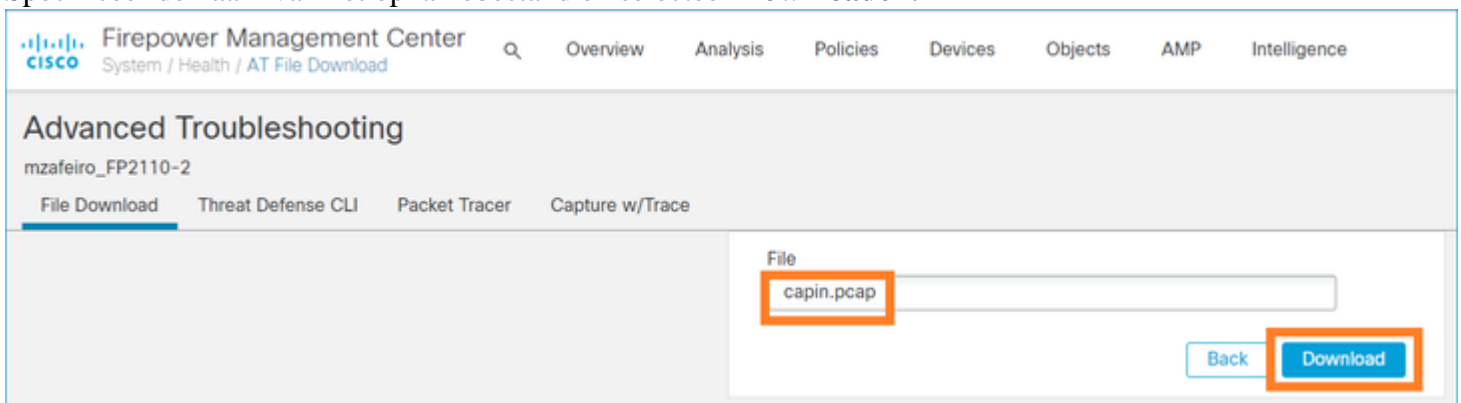


Stap 4

Selecteer **Geavanceerde probleemoplossing**:



Specificeer de naam van het opnamebestand en selecteer **Downloaden**:

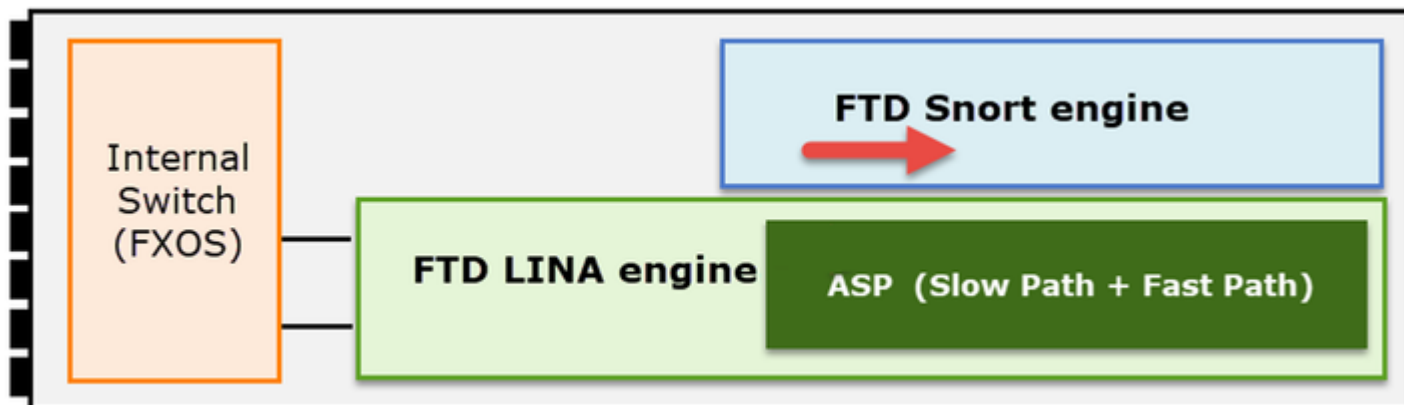


Voor meer voorbeelden hoe u opnamen kunt in- of verzamelen vanuit de FMC UI, controleer u dit document:

<https://www.cisco.com/c/en/us/support/docs/security/firepower-ngfw/212474-working-with-firepower-threat-defense-f.html>

FTD-snortopnamen inschakelen en verzamelen

Het opnamepunt wordt hier in het beeld weergegeven.



Snelniveaumeting inschakelen:

```
<#root>
```

```
>
```

```
capture-traffic
```

```
Please choose domain to capture traffic from:
```

```
0 - br1
```

```
1 - Router
```

```
Selection?
```

```
1
```

```
Please specify tcpdump options desired.
```

```
(or enter '?' for a list of supported options)
```

```
Options:
```

```
-n host 192.168.101.1
```

U kunt de opname als volgt naar een bestand met de naam capture.pcap schrijven en via FTP naar een externe server kopiëren:

```
<#root>
```

```
>
```

```
capture-traffic
```

```
Please choose domain to capture traffic from:
```

```
0 - br1
```

```
1 - Router
```

```
Selection?
```

```
1
```

```
Please specify tcpdump options desired.
```

```
(or enter '?' for a list of supported options)
```


Options:

```
-w capture.pcap host 192.168.101.1
```

```
CTRL + C <- to stop the capture
```

```
>
```

```
file copy 10.229.22.136 ftp / capture.pcap
```

```
Enter password for ftp@10.229.22.136:
```

```
Copying capture.pcap
```

```
Copy successful.
```

```
>
```

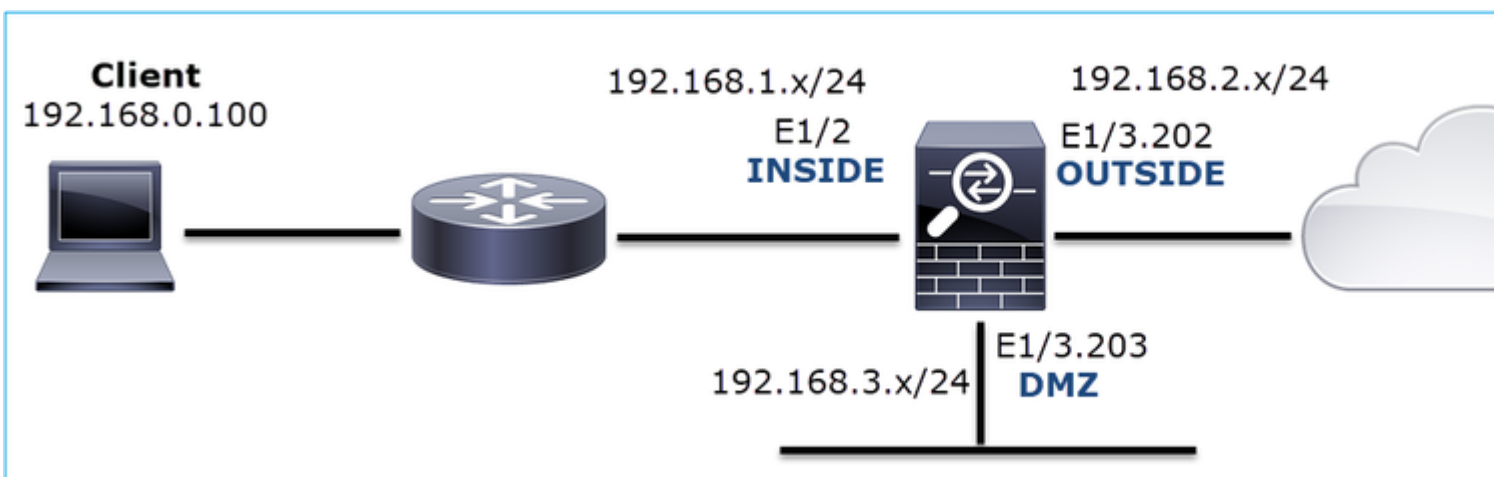
Controleer dit document voor meer voorbeelden van vastlegging op kortniveau die verschillende opnamefilters omvatten:

<https://www.cisco.com/c/en/us/support/docs/security/firepower-ngfw/212474-working-with-firepower-threat-defense-f.html>

Problemen oplossen

Situatie 1. Geen TCP/SYN op uitgaande interface

De topologie wordt hier in het beeld getoond:



Probleembeschrijving: HTTP werkt niet

Beïnvloede stroom:

SRC IP: 192.168.0.10

Laatste IP: 10.10.1.100

Protocol: TCP 80

Capture Analysis

Opnamen op de FTD LINA-motor inschakelen:

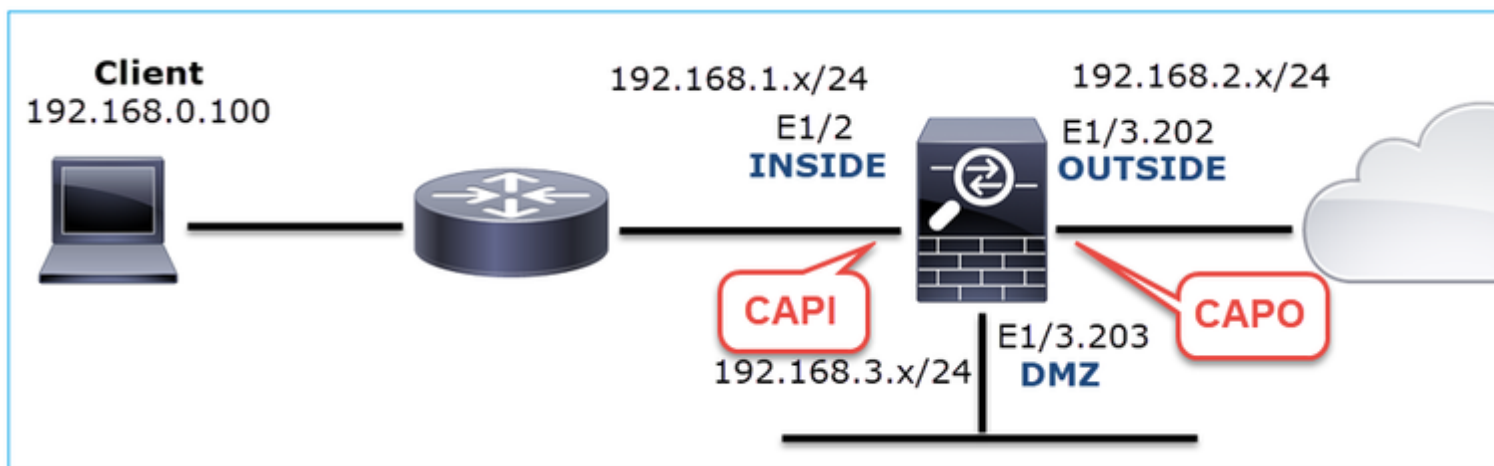
```
<#root>
```

```
firepower#
```

```
capture CAPI int INSIDE match ip host 192.168.0.100 host 10.10.1.100
```

```
firepower#
```

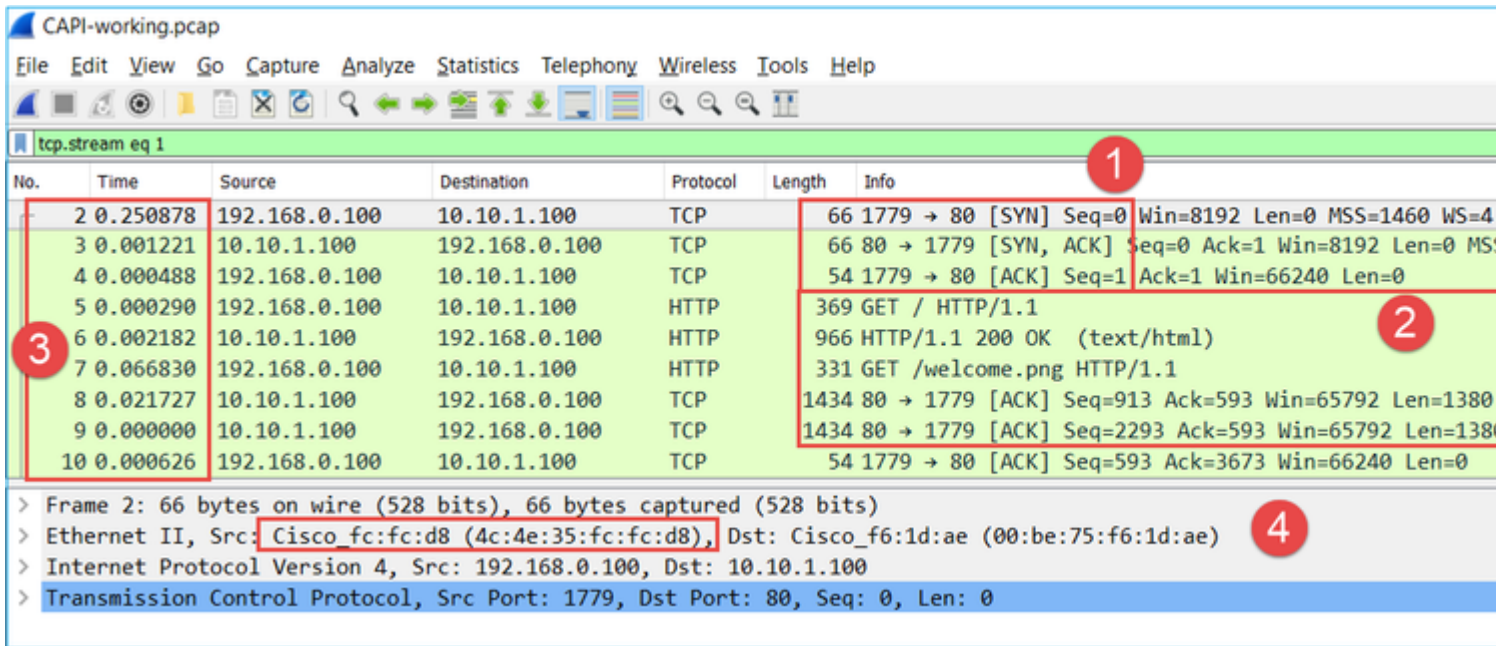
```
capture CAPO int OUTSIDE match ip host 192.168.0.100 host 10.10.1.100
```



Opname - Functioneel scenario:

Als basislijn is het altijd erg handig om opnamen te maken van een functioneel scenario.

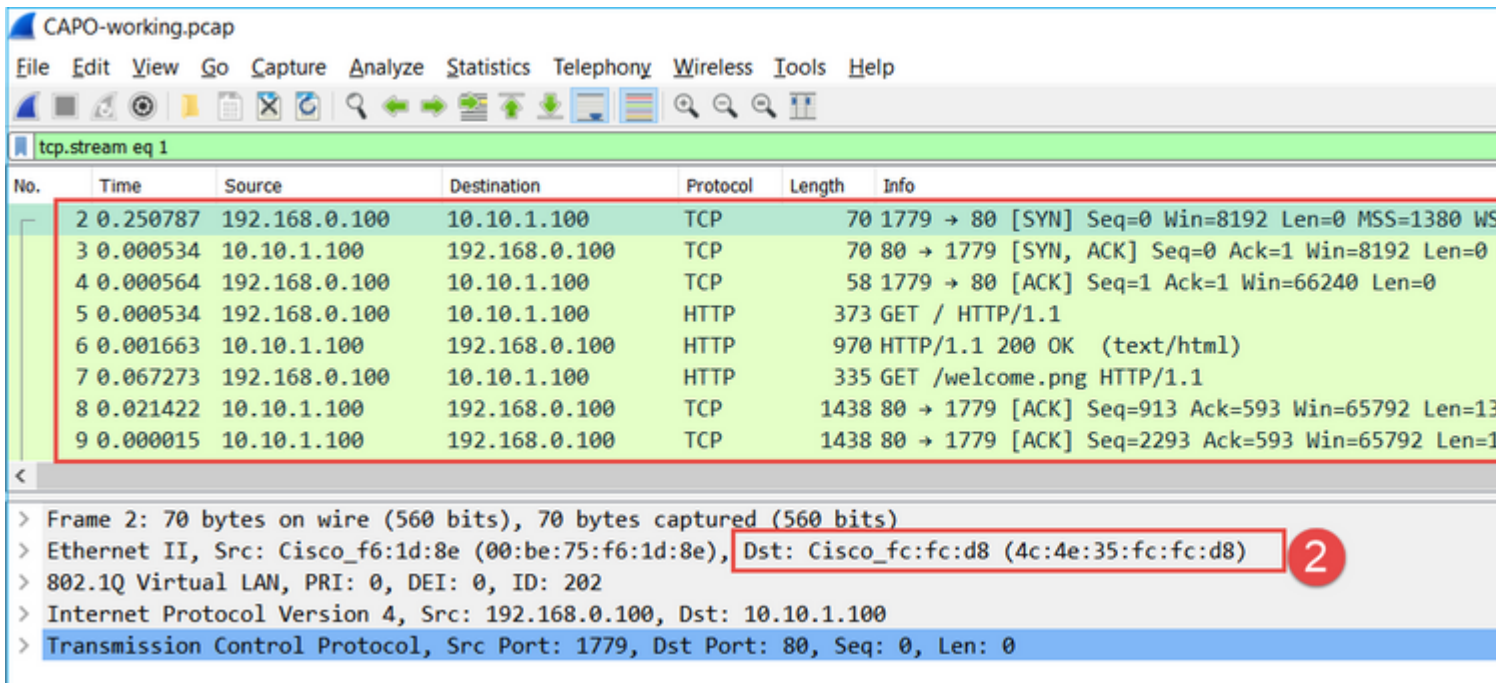
Capture genomen op NGFW INSIDE interface, is zoals in het beeld:



Belangrijkste punten:

1. TCP 3-weg handdruk.
2. Bidirectionele gegevensuitwisseling.
3. Geen vertragingen tussen de pakketten (gebaseerd op het tijdsverschil tussen de pakketten)
4. Source MAC is het juiste downstream apparaat.

Capture genomen op NGFW BUITEN interface, wordt hier in het beeld getoond:



Belangrijkste punten:

1. Dezelfde gegevens als bij de CAPI-vastlegging.
2. Doelstelling MAC is het juiste upstream apparaat.

Captures - niet-functioneel scenario

Van het apparaat CLI zien de opnamen er zo uit:

```
<#root>
firepower#
show capture

capture CAPI type raw-data interface INSIDE
[Capturing - 484 bytes]

  match ip host 192.168.0.100 host 10.10.1.100
capture CAPO type raw-data interface OUTSIDE
[Capturing - 0 bytes]

  match ip host 192.168.0.100 host 10.10.1.100
```

CAPI inhoud:

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

6 packets captured

  1: 11:47:46.911482  192.168.0.100.3171 > 10.10.1.100.80:
s
1089825363:1089825363(0) win 8192 <mss 1460,nop,wscale 2,nop,nop,sackOK>
  2: 11:47:47.161902  192.168.0.100.3172 > 10.10.1.100.80:
s
3981048763:3981048763(0) win 8192 <mss 1460,nop,wscale 2,nop,nop,sackOK>
  3: 11:47:49.907683  192.168.0.100.3171 > 10.10.1.100.80:
s
1089825363:1089825363(0) win 8192 <mss 1460,nop,wscale 2,nop,nop,sackOK>
  4: 11:47:50.162757  192.168.0.100.3172 > 10.10.1.100.80:
s
3981048763:3981048763(0) win 8192 <mss 1460,nop,wscale 2,nop,nop,sackOK>
  5: 11:47:55.914640  192.168.0.100.3171 > 10.10.1.100.80:
s
1089825363:1089825363(0) win 8192 <mss 1460,nop,nop,sackOK>
  6: 11:47:56.164710  192.168.0.100.3172 > 10.10.1.100.80:
s
3981048763:3981048763(0) win 8192 <mss 1460,nop,nop,sackOK>
```

```

<#root>

firepower#

show capture CAPO

0 packet captured

0 packet shown

```

Dit is het beeld van CAPI Capture in Wireshark:

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.0.100	10.10.1.100	TCP	66	3171 → 80 [SYN] Seq=0 Win=8192 Len=0
2	0.250420	192.168.0.100	10.10.1.100	TCP	66	3172 → 80 [SYN] Seq=0 Win=8192 Len=0
3	2.745781	192.168.0.100	10.10.1.100	TCP	66	[TCP Retransmission] 3171 → 80 [SYN] Seq=0 Win=8192 Len=0
4	0.255074	192.168.0.100	10.10.1.100	TCP	66	[TCP Retransmission] 3172 → 80 [SYN] Seq=0 Win=8192 Len=0
5	5.751883	192.168.0.100	10.10.1.100	TCP	62	[TCP Retransmission] 3171 → 80 [SYN] Seq=0 Win=8192 Len=0
6	0.250070	192.168.0.100	10.10.1.100	TCP	62	[TCP Retransmission] 3172 → 80 [SYN] Seq=0 Win=8192 Len=0

> Frame 1: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface
 > Ethernet II, Src: Cisco_fc:fc:d8 (4c:4e:35:fc:fc:d8), Dst: Cisco_f6:1d:ae (00:be:75:f6:1d:ae)
 > Internet Protocol Version 4, Src: 192.168.0.100, Dst: 10.10.1.100
 > Transmission Control Protocol, Src Port: 3171, Dst Port: 80, Seq: 0, Len: 0

Belangrijkste punten:

1. Alleen TCP SYN-pakketten worden gezien (geen TCP 3-weg handshake).
2. Er zijn 2 TCP-sessies (bronpoort 3171 en 3172) die niet kunnen worden ingesteld. De bronclient verstuurt de TCP/SYN-pakketten opnieuw. Deze opnieuw verzonden pakketten worden geïdentificeerd door de Wireshark als TCP-wederuitzendingen.
3. De TCP-heruitzendingen gebeuren elke ~3 en dan 6 etc seconden.
4. Het MAC-adres van de bron is afkomstig van het juiste downstream-apparaat.

Op basis van de 2 captures kan worden geconcludeerd dat:

- Een pakket van een specifieke 5-tuple (src/dst IP, src/dst poort, protocol) wordt ontvangen door de firewall op de verwachte interface (INSIDE).
- Een pakket verlaat niet de firewall op de verwachte interface (BUITEN).

Aanbevolen acties

De acties die in deze paragraaf worden opgesomd, hebben tot doel de kwestie verder af te zwakken.

Actie 1. Controleer het spoor van een geëmuleerd pakket.

Gebruik het packet-tracer hulpmiddel om te zien hoe een pakket verondersteld wordt om door de firewall behandeld te worden. Indien het pakket wordt gedropt door het firewall-toegangsbeleid, ziet het spoor van het geëmuleerde pakket er ongeveer hetzelfde uit als deze uitvoer:

```

<#root>

firepower#

```

```
packet-tracer input INSIDE tcp 192.168.0.100 11111 10.10.1.100 80
```

```
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: ROUTE-LOOKUP  
Subtype: Resolve Egress Interface  
Result: ALLOW  
Config:  
Additional Information:  
found next-hop 192.168.2.72 using egress ifc OUTSIDE
```

```
Phase: 4
```

```
Type: ACCESS-LIST
```

```
Subtype: log
```

```
Result: DROP
```

```
Config:  
access-group CSM_FW_ACL_ global  
access-list CSM_FW_ACL_ advanced deny ip any any rule-id 268439946 event-log flow-start  
access-list CSM_FW_ACL_ remark rule-id 268439946: ACCESS POLICY: FTD_Policy - Default  
access-list CSM_FW_ACL_ remark rule-id 268439946: L4 RULE: DEFAULT ACTION RULE  
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, Drop-location: frame 0x00005647a4f4b120 flow
```

Actie 2. Controleer de sporen van levende pakketten.

Laat het pakketspoor toe om te controleren hoe de echte TCP/SYN-pakketten door de firewall worden verwerkt. Standaard worden alleen de eerste 50 ingangspakketten overgetrokken:

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

Schakel de opnamebuffer uit:

```
<#root>
firepower#
clear capture /all
```

Als het pakket wordt losgelaten door het firewall-toegangsbeleid, ziet het spoor er ongeveer hetzelfde uit als deze uitvoer:

```
<#root>
firepower#
show capture CAPI packet-number 1 trace

6 packets captured

  1: 12:45:36.279740      192.168.0.100.3630 > 10.10.1.100.80: S 2322685377:2322685377(0) win 8192 <ms
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: ROUTE-LOOKUP
Subtype: Resolve Egress Interface
Result: ALLOW
Config:
Additional Information:
found next-hop 192.168.2.72 using egress ifc  OUTSIDE

Phase: 4
Type: ACCESS-LIST
Subtype: log
```

Result: DROP

Config:

```
access-group CSM_FW_ACL_ global
access-list CSM_FW_ACL_ advanced deny ip any any rule-id 268439946 event-log flow-start
access-list CSM_FW_ACL_ remark rule-id 268439946: ACCESS POLICY: FTD_Policy - Default
access-list CSM_FW_ACL_ remark rule-id 268439946: L4 RULE: DEFAULT ACTION RULE
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, Drop-location: frame 0x00005647a4f4b120 flow

1 packet shown

Actie 3. Controleer FTD Lina logs.

Controleer dit document om Syslog op FTD via FMC te configureren:

<https://www.cisco.com/c/en/us/support/docs/security/firepower-ngfw/200479-Configure-Logging-on-FTD-via-FMC.html>

Het is sterk aanbevolen om een externe Syslog server geconfigureerd te hebben voor FTD Lina logs. Als er geen externe Syslog-server geconfigureerd is, activeert u lokale bufferlogbestanden op de firewall terwijl u problemen oplost. De logconfiguratie die in dit voorbeeld wordt getoond, is een goed beginpunt:

```
<#root>
```

```
firepower#
```

```
show run logging
```

```
â€|
logging enable
logging timestamp
logging buffer-size 1000000
logging buffered informational
```

Stel de terminal pager in op 24 lijnen om de terminal pager te besturen:

```
<#root>
```

```
firepower#
```


Schakel de opnamebuffer uit:

```
<#root>
firepower#
clear logging buffer
```

Test de verbinding en controleer de logbestanden met een parser filter. In dit voorbeeld worden de pakketten verwijderd door het beleid voor firewalltoegang:

```
<#root>
firepower#
show logging | include 10.10.1.100
Oct 09 2019 12:55:51: %FTD-4-106023: Deny tcp src INSIDE:192.168.0.100/3696 dst OUTSIDE:10.10.1.100/80 b
Oct 09 2019 12:55:51: %FTD-4-106023: Deny tcp src INSIDE:192.168.0.100/3697 dst OUTSIDE:10.10.1.100/80 b
Oct 09 2019 12:55:54: %FTD-4-106023: Deny tcp src INSIDE:192.168.0.100/3696 dst OUTSIDE:10.10.1.100/80 b
Oct 09 2019 12:55:54: %FTD-4-106023: Deny tcp src INSIDE:192.168.0.100/3697 dst OUTSIDE:10.10.1.100/80 b
```

Actie 4. Controleer of de firewall ASP daalt.

Als u vermoedt dat het pakket door de firewall wordt gedropt, kunt u de tellers zien van alle pakketten die door de firewall op softwareniveau worden gedropt:

```
<#root>
firepower#
show asp drop

Frame drop:
  No route to host (no-route)                234
  Flow is denied by configured rule (acl-drop) 71

Last clearing: 07:51:52 UTC Oct 10 2019 by enable_15

Flow drop:

Last clearing: 07:51:52 UTC Oct 10 2019 by enable_15
```

U kunt opnamen inschakelen om alle ASP-softwareniveau-dalingen te zien:

```
<#root>
```

```
firepower#
```

```
capture ASP type asp-drop all buffer 33554432 headers-only
```

Tip: Als u niet geïnteresseerd bent in de pakketinhoud kunt u alleen de pakketheader opnemen (optie alleen met kopregels). Dit staat u toe om veel meer pakketten in de vangstbuffer te vangen. Daarnaast kunt u de omvang van de opnamebuffer (standaard is 500Kbytes) vergroten tot een waarde van 32 Mbytes (bufferoptie). Ten slotte kunt u vanaf FTD versie 6.3 met de optie bestandsgrootte een opnamebestand tot 10 GBytes configureren. In dat geval kunt u de opnameinhoud alleen in een pcap-formaat zien.

Om de opnameinhoud te controleren, kunt u een filter gebruiken om uw zoekopdracht te verfijnen:

```
<#root>
```

```
firepower#
```

```
show capture ASP | include 10.10.1.100
```

```
18: 07:51:57.823672 192.168.0.100.12410 > 10.10.1.100.80: S 1870382552:1870382552(0) win 8192 <mss 1
19: 07:51:58.074291 192.168.0.100.12411 > 10.10.1.100.80: S 2006489005:2006489005(0) win 8192 <mss 1
26: 07:52:00.830370 192.168.0.100.12410 > 10.10.1.100.80: S 1870382552:1870382552(0) win 8192 <mss 1
29: 07:52:01.080394 192.168.0.100.12411 > 10.10.1.100.80: S 2006489005:2006489005(0) win 8192 <mss 1
45: 07:52:06.824282 192.168.0.100.12410 > 10.10.1.100.80: S 1870382552:1870382552(0) win 8192 <mss 1
46: 07:52:07.074230 192.168.0.100.12411 > 10.10.1.100.80: S 2006489005:2006489005(0) win 8192 <mss 1
```

In dit geval, omdat de pakketten al op interfaceniveau worden overgetrokken, wordt de reden voor de daling niet vermeld in de ASP-opname. Herinner dat een pakket slechts op één plaats (ingangsinterface of de daling van het ASPIS) kan worden gevonden. In dat geval is het raadzaam om meerdere ASP-druppels te nemen en een specifieke ASP-reden in te stellen. Hier is een aanbevolen benadering:

1. Schakel de huidige ASP-valtellers uit:

```
<#root>
```

```
firepower#
```

```
clear asp drop
```

2. Verzend de stroom die u door de firewall problemen oplost (voer een test uit).

3. Controleer nogmaals de ASP drop tellers en noteer de verhoogde.

```
<#root>
```

```
firepower#
```

```
show asp drop
```

```
Frame drop:
```

```
No route to host (
```

```

no-route
)
Flow is denied by configured rule (
acl-drop
)
71
234

```

4. Schakel ASP Capture(s) in voor de specifieke druppels die worden gezien:

```

<#root>
firepower#
capture ASP_NO_ROUTE type asp-drop no-route
firepower#
capture ASP_ACL_DROP type asp-drop acl-drop

```

5. Verzend de stroom die u problemen oplost door de firewall (voer een test uit).

6. Controleer of het ASP-bestand wordt opgenomen. In dit geval, werden de pakketten gelaten vallen toe te schrijven aan een ontbrekende route:

```

<#root>
firepower#
show capture ASP_NO_ROUTE | include 192.168.0.100.*10.10.1.100
93: 07:53:52.381663 192.168.0.100.12417 > 10.10.1.100.80: S 3451917925:3451917925(0) win 8192 <mss 1
95: 07:53:52.632337 192.168.0.100.12418 > 10.10.1.100.80: S 1691844448:1691844448(0) win 8192 <mss 1
101: 07:53:55.375392 192.168.0.100.12417 > 10.10.1.100.80: S 3451917925:3451917925(0) win 8192 <mss 1
102: 07:53:55.626386 192.168.0.100.12418 > 10.10.1.100.80: S 1691844448:1691844448(0) win 8192 <mss 1
116: 07:54:01.376231 192.168.0.100.12417 > 10.10.1.100.80: S 3451917925:3451917925(0) win 8192 <mss 1
117: 07:54:01.626310 192.168.0.100.12418 > 10.10.1.100.80: S 1691844448:1691844448(0) win 8192 <mss 1

```

Actie 5. Controleer de verbindingstabel met de FTD-lijn.

Er kunnen gevallen zijn waarin je verwacht dat het pakket interface 'X' zal verlaten, maar om welke redenen dan ook wordt interface 'Y' eruit gelicht. De vaststelling van de interface voor het verlaten van de firewall is gebaseerd op deze volgorde van werking:

1. Vaste verbinding zoeken
2. Network Address Translation (NAT) lookup - UN-NAT (bestemming NAT) fase krijgt voorrang boven PBR en routerlookup.
3. Op beleid gebaseerde routing (PBR)
4. Routing-tabel

U kunt de FTD-verbindingstabel als volgt controleren:

<#root>

firepower#

show conn

2 in use, 4 most used

Inspect Snort:

 preserve-connection: 2 enabled, 0 in effect, 4 most enabled, 0 most in effect

TCP

DMZ

10.10.1.100:

80

INSIDE

192.168.0.100:

11694

, idle 0:00:01, bytes 0, flags

aA N1

TCP

DMZ

10.10.1.100:80

INSIDE

192.168.0.100:

11693

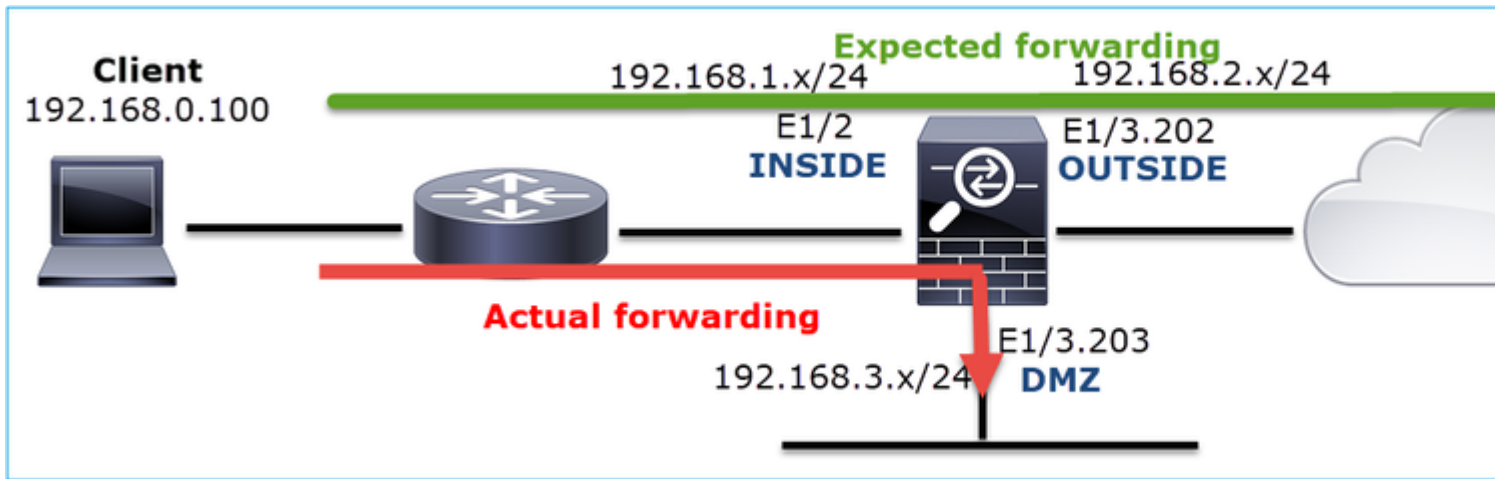
, idle 0:00:01, bytes 0, flags

aA N1

Belangrijkste punten:

- Gebaseerd op de vlaggen (Aa) is de verbinding embryonaal (half geopend - alleen TCP SYN werd gezien door de firewall).
- Gebaseerd op de bron/bestemmingshavens is de toegangsinterface BINNENIN en de uitgangsinterface is DMZ.

Dit kan hier worden gevisualiseerd:



Opmerking: Aangezien alle FTD-interfaces een beveiligingsniveau van 0 hebben, is de interfacevolgorde in de **show conn**-uitvoer gebaseerd op het interfacenummer. Met name de interface met een hoger vpif-nummer (Virtual Platform Interface Number) wordt als binnenin geselecteerd, terwijl de interface met een lager vpif-nummer als buitenkant is geselecteerd. U kunt de interface vpif waarde met het bevel van het **showinterfacedetail** zien. Verwante verbetering, Cisco bug-id [CSCvi15290](https://www.cisco.com/c/enus/bugtools/bugtools/bugtools.html?bugid=CSCvi15290) ENH: FTD toont de verbindingdirectionaliteit in FTD 'toon conn' uitvoer

```
<#root>
firepower#
show interface detail | i Interface number is|Interface [P|E].*is up
...
Interface Ethernet1/2 "INSIDE", is up, line protocol is up
  Interface number is
19
Interface Ethernet1/3.202 "OUTSIDE", is up, line protocol is up
  Interface number is
20
Interface Ethernet1/3.203 "DMZ", is up, line protocol is up
  Interface number is
22
```

Opmerking: Vanaf FirePOWER-software release 6.5, ASA release 9.13.x, de **show conn long** en **toon conn detail** commando outputs verstrekken informatie over de verbinding initiator en responder

Output 1:

```
<#root>
firepower#
show conn long
...
```

TCP OUTSIDE: 192.168.2.200/80 (192.168.2.200/80) INSIDE: 192.168.1.100/46050 (192.168.1.100/46050), flags

Initiator: 192.168.1.100, Responder: 192.168.2.200

Connection lookup keyid: 228982375

Output 2:

```
<#root>
```

```
firepower#
```

```
show conn detail
```

```
...
```

```
TCP OUTSIDE: 192.168.2.200/80 INSIDE: 192.168.1.100/46050,  
  flags aA N1, idle 4s, uptime 11s, timeout 30s, bytes 0
```

Initiator: 192.168.1.100, Responder: 192.168.2.200

Connection lookup keyid: 228982375

Bovendien toont de **show conn long** de NATed IPs binnen een haakje in het geval van een netwerkadresomzetting:

```
<#root>
```

```
firepower#
```

```
show conn long
```

```
...
```

```
TCP OUTSIDE: 192.168.2.222/80 (192.168.2.222/80) INSIDE: 192.168.1.100/34792 (192.168.2.150/34792), flags  
  Initiator: 192.168.1.100, Responder: 192.168.2.222  
  Connection lookup keyid: 262895
```

Actie 6. Controleer het ARP-cache (Firewall Address Resolution Protocol).

Als de firewall de volgende hop niet kan oplossen, laat de firewall stilletjes het oorspronkelijke pakket vallen (in dit geval TCP/SYN) en verstuurt voortdurend ARP-verzoeken tot het de volgende hop oplost.

Gebruik de opdracht om de ARP-cache van de firewall te zien:

```
<#root>
```

```
firepower#
```

```
show arp
```

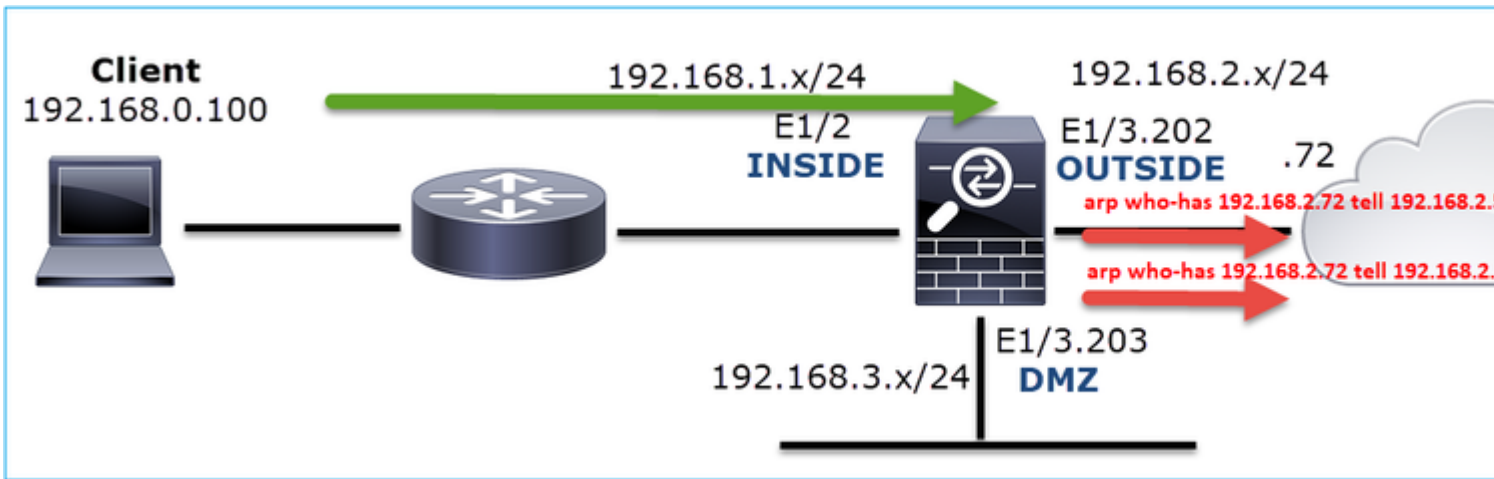
Bovendien, om te controleren of er onopgeloste hosts zijn kunt u de opdracht gebruiken:

```
<#root>
firepower#
  show arp statistics
      Number of ARP entries in ASA: 0
      Dropped blocks in ARP: 84
      Maximum Queued blocks: 3
      Queued blocks: 0
      Interface collision ARPs Received: 0
      ARP-defense Gratuitous ARPS sent: 0
      Total ARP retries:
182          < indicates a possible issue for some hosts
      Unresolved hosts:
1
< this is the current status
      Maximum Unresolved hosts: 2
```

Als u de ARP-handeling verder wilt controleren, kunt u een ARP-specifieke opname inschakelen:

```
<#root>
firepower#
capture ARP ethernet-type arp interface OUTSIDE
firepower#
show capture ARP
...
 4: 07:15:16.877914      802.1Q vlan#202 P0 arp
who-has 192.168.2.72 tell 192.168.2.50
 5: 07:15:18.020033      802.1Q vlan#202 P0 arp who-has 192.168.2.72 tell 192.168.2.50
```

In deze output, probeert de firewall (192.168.2.50) om de volgende-hop (192.168.2.72) op te lossen, maar er is geen ARP antwoord



De output toont hier een functioneel scenario met juiste ARP resolutie:

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

2 packets captured

  1: 07:17:19.495595      802.1Q vlan#202 P0
arp who-has 192.168.2.72 tell 192.168.2.50

  2: 07:17:19.495946      802.1Q vlan#202 P0
arp reply 192.168.2.72 is-at 4c:4e:35:fc:fc:d8

2 packets shown
```

```
<#root>
firepower#
show arp

    INSIDE 192.168.1.71 4c4e.35fc.fcd8 9
    OUTSIDE 192.168.2.72 4c4e.35fc.fcd8 9
```

Als er geen ARP-ingang is, wordt een spoor van een levend TCP/SYN-pakket weergegeven:

```
<#root>
firepower#
show capture CAPI packet-number 1 trace

6 packets captured

  1: 07:03:43.270585
```


192.168.0.100.11997 > 10.10.1.100.80

: S 4023707145:4023707145(0) win 8192 <mss 1460,nop,wscale 2,nop,nop,sackOK>

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: ROUTE-LOOKUP

Subtype: Resolve Egress Interface

Result: ALLOW

Config:

Additional Information:

found next-hop 192.168.2.72 using egress ifc OUTSIDE

â€¦

Phase: 14

Type: FLOW-CREATION

Subtype:

Result: ALLOW

Config:

Additional Information:

New flow created with id 4814, packet dispatched to next module

â€¦

Phase: 17

Type: ROUTE-LOOKUP

Subtype: Resolve Egress Interface

Result: ALLOW

Config:

Additional Information:

found next-hop 192.168.2.72 using egress ifc OUTSIDE

Result:

input-interface: INSIDE

input-status: up

input-line-status: up

output-interface: OUTSIDE

output-status: up

output-line-status: up

Action: allow

Zoals in de output kan worden gezien, toont het spoor **Actie: sta toe** zelfs wanneer de volgende hop niet bereikbaar is en het pakket stil door de firewall wordt gelaten vallen! In dit geval moet ook het pakkettraceergereedschap worden gecontroleerd, omdat dit een nauwkeurigere uitvoer oplevert:

<#root>

firepower#

packet-tracer input INSIDE tcp 192.168.0.100 1111 10.10.1.100 80

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: ROUTE-LOOKUP

Subtype: Resolve Egress Interface

Result: ALLOW

Config:

Additional Information:

found next-hop 192.168.2.72 using egress ifc OUTSIDE

â€¦

Phase: 14

Type: FLOW-CREATION

Subtype:

Result: ALLOW

Config:

Additional Information:

New flow created with id 4816, packet dispatched to next module

â€¦

Phase: 17

Type: ROUTE-LOOKUP

Subtype: Resolve Egress Interface

Result: ALLOW

Config:

Additional Information:

found next-hop 192.168.2.72 using egress ifc OUTSIDE

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: (no-v4-adjacency) No valid V4 adjacency, Drop-location: frame 0x00005647a4e86109 flow (NA)/

In recente versies van ASA/Firepower is het vorige bericht geoptimaliseerd om:

<#root>

Drop-reason: (no-v4-adjacency) No valid V4 adjacency.

Check ARP table (show arp) has entry for nexthop

., Drop-location: f

Mogelijke oorzaken en aanbevolen acties Samenvatting

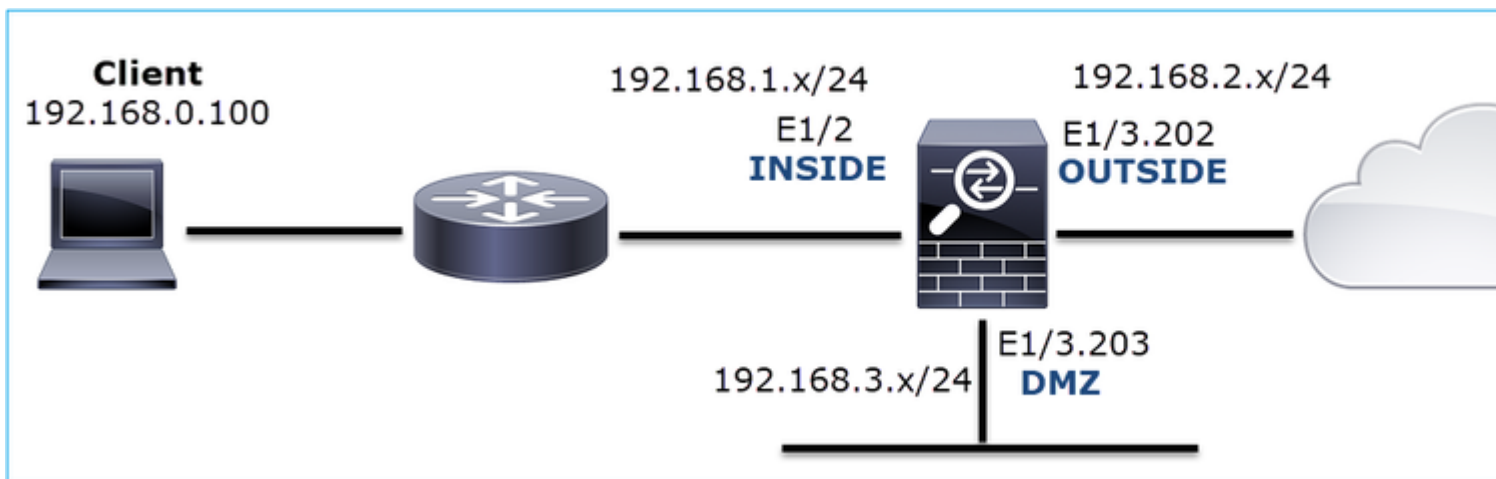
Als u alleen een TCP SYN-pakket op de toegangsinterfaces ziet, maar geen TCP SYN-pakket verzonden uit de verwachte uitgangsinterface, zijn enkele mogelijke oorzaken:

Mogelijke oorzaak	Aanbevolen acties
Het pakket wordt verbroken door het beleid voor firewalltoegang.	<ul style="list-style-type: none">• Gebruik packet-tracer of opname met trace om te zien hoe de firewall het pakket verwerkt.• Controleer de firewalllogboeken.• Controleer of de firewall ASP druppels (toon een asp drop of opname type asp-drop).• Controleer FMC Connection-gebeurtenissen. Dit veronderstelt dat de regel loggen ingeschakeld heeft.
Het opnamefilter is onjuist.	<ul style="list-style-type: none">• Gebruik pakkettracer of leg w/trace op om te zien of er NAT-vertaling is die de bron- of bestemmings-IP wijzigt. Pas in dat geval uw opnamefilter aan.• toon conn lange opdrachtoutput de NATed IP's toont.
Het pakket wordt verzonden naar een andere uitgangsinterface.	<ul style="list-style-type: none">• Gebruik packet-tracer of opname met trace om te zien hoe de firewall met het pakket omgaat. Herinner de orde van verrichtingen die de bepaling van de uitgangsinterface, huidige verbinding, UN-NAT, PBR en het Verpletteren van tabelraadpleging beschouwen.• Controleer de firewalllogboeken.• Controleer de tabel met de firewallverbinding (toon verbinding). <p>Als het pakket naar een verkeerde interface wordt verzonden omdat het een huidige verbinding aanpast gebruik het bevel duidelijke conn adres en specificeer 5 -tuple van de verbinding die u wilt ontruimen.</p>
Er is geen route naar de bestemming.	<ul style="list-style-type: none">• Gebruik packet-tracer of opname met trace om te zien hoe de firewall het pakket verwerkt.• Controleer of de firewall ASP druppels (toon een asp drop) voor geen-route drop reden.

Er is geen ARP ingang op de uitgangside.	<ul style="list-style-type: none"> Controleer de firewall ARP cache (arp weergeven). Gebruik packet-tracer om te zien of er een geldige nabijheid is.
De uitgangside is down.	Controleer de uitvoer van de ip -opdracht met de interface van de show op de firewall en controleer de interfacestatus.

Situatie 2. TCP/SYN vanaf client, TCP/RST vanaf server

Dit beeld toont de topologie:



Probleembeschrijving: HTTP werkt niet

Beïnvloede stroom:

SRC IP: 192.168.0.10

Laatste IP: 10.10.1.100

Protocol: TCP 80

Capture Analysis

Schakel opnamen in op de FTD LINA engine.

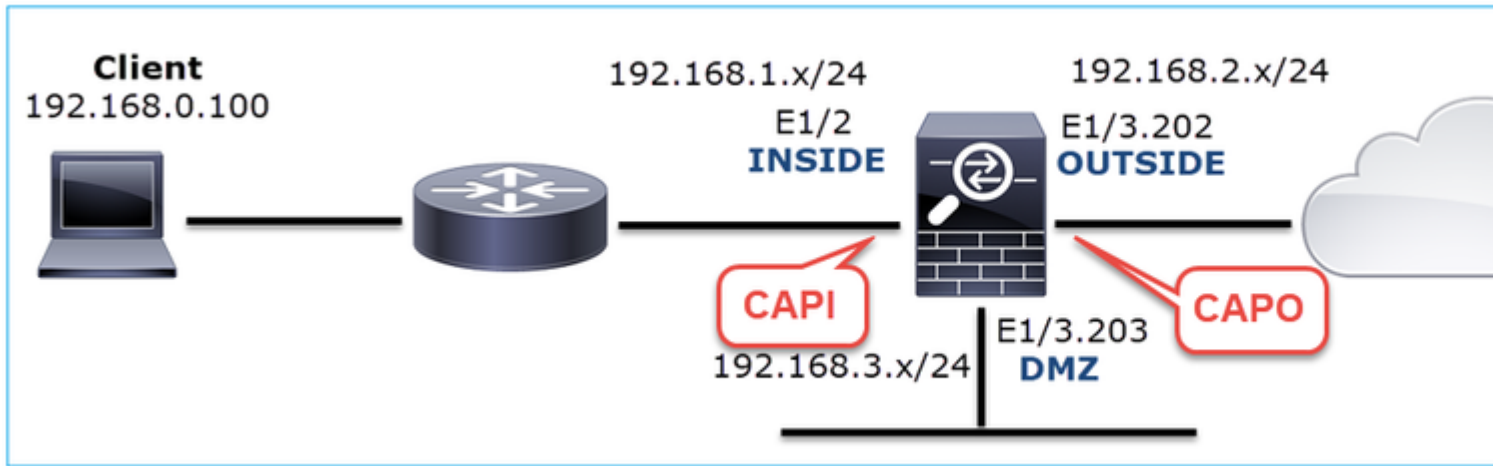
```
<#root>
```

```
firepower#
```

```
capture CAPI int INSIDE match ip host 192.168.0.100 host 10.10.1.100
```

```
firepower#
```

```
capture CAPO int OUTSIDE match ip host 192.168.0.100 host 10.10.1.100
```



Captures - niet-functioneel scenario:

Van het apparaat CLI zien de opnamen er als volgt uit:

```
<#root>
firepower#
show capture
capture CAPI type raw-data trace interface INSIDE [Capturing -
834 bytes
]
match ip host 192.168.0.100 host 10.10.1.100
capture CAPO type raw-data interface OUTSIDE [Capturing -
878 bytes
]
match ip host 192.168.0.100 host 10.10.1.100
```

CAPI inhoud:

```
<#root>
firepower#
show capture CAPI
1: 05:20:36.654217 192.168.0.100.22195 > 10.10.1.100.80:
S
1397289928:1397289928(0) win 8192 <mss 1460,nop,wscale 2,nop,nop,sackOK>
2: 05:20:36.904311 192.168.0.100.22196 > 10.10.1.100.80:
S
2171673258:2171673258(0) win 8192 <mss 1460,nop,wscale 2,nop,nop,sackOK>
3: 05:20:36.905043 10.10.1.100.80 > 192.168.0.100.22196:
R
```

```
1850052503:1850052503(0) ack 2171673259 win 0
 4: 05:20:37.414132 192.168.0.100.22196 > 10.10.1.100.80:
```

S

```
2171673258:2171673258(0) win 8192 <mss 1460,nop,wscale 2,nop,nop,sackOK>
 5: 05:20:37.414803 10.10.1.100.80 > 192.168.0.100.22196:
```

R

```
31997177:31997177(0) ack 2171673259 win 0
 6: 05:20:37.914183 192.168.0.100.22196 > 10.10.1.100.80:
```

S

```
2171673258:2171673258(0) win 8192 <mss 1460,nop,nop,sackOK>
...
```

CAPO inhoud:

<#root>

firepower#

show capture CAPO

```
 1: 05:20:36.654507 802.1Q vlan#202 P0 192.168.0.100.22195 > 10.10.1.100.80:
```

S

```
2866789268:2866789268(0) win 8192 <mss 1380,nop,wscale 2,nop,nop,sackOK>
 2: 05:20:36.904478 802.1Q vlan#202 P0 192.168.0.100.22196 > 10.10.1.100.80:
```

S

```
4785344:4785344(0) win 8192 <mss 1380,nop,wscale 2,nop,nop,sackOK>
 3: 05:20:36.904997 802.1Q vlan#202 P0 10.10.1.100.80 > 192.168.0.100.22196:
```

R

```
0:0(0) ack 4785345 win 0
 4: 05:20:37.414269 802.1Q vlan#202 P0 192.168.0.100.22196 > 10.10.1.100.80:
```

S

```
4235354730:4235354730(0) win 8192 <mss 1380,nop,wscale 2,nop,nop,sackOK>
 5: 05:20:37.414758 802.1Q vlan#202 P0 10.10.1.100.80 > 192.168.0.100.22196:
```

R

```
0:0(0) ack 4235354731 win 0
 6: 05:20:37.914305 802.1Q vlan#202 P0 192.168.0.100.22196 > 10.10.1.100.80:
```

S

```
4118617832:4118617832(0) win 8192 <mss 1380,nop,nop,sackOK>
```

Deze afbeelding toont de opname van CAPI in Wireshark.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.0.100	10.10.1.100	TCP	66	22195 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460
2	0.250094	192.168.0.100	10.10.1.100	TCP	66	22196 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460
3	0.000732	10.10.1.100	192.168.0.100	TCP	54	80 → 22196 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
4	0.509089	192.168.0.100	10.10.1.100	TCP	62	[TCP Retransmission] 22196 → 80 [SYN] Seq=0 Win=0 Len=0
5	0.000671	10.10.1.100	192.168.0.100	TCP	54	80 → 22196 [RST, ACK] Seq=2476911971 Ack=1 Win=0 Len=0
6	0.499380	192.168.0.100	10.10.1.100	TCP	62	[TCP Retransmission] 22196 → 80 [SYN] Seq=0 Win=0 Len=0
7	0.000625	10.10.1.100	192.168.0.100	TCP	54	80 → 22196 [RST, ACK] Seq=2853655305 Ack=1 Win=0 Len=0
8	1.739729	192.168.0.100	10.10.1.100	TCP	66	[TCP Retransmission] 22195 → 80 [SYN] Seq=0 Win=0 Len=0
9	0.000611	10.10.1.100	192.168.0.100	TCP	54	80 → 22195 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
10	0.499385	192.168.0.100	10.10.1.100	TCP	62	[TCP Retransmission] 22195 → 80 [SYN] Seq=0 Win=0 Len=0
11	0.000671	10.10.1.100	192.168.0.100	TCP	54	80 → 22195 [RST, ACK] Seq=151733665 Ack=1 Win=0 Len=0

> Frame 1: 66 bytes on wire (528 bits), 66 bytes captured (528 bits)
 > Ethernet II, Src: Cisco_fc:fc:d8 (4c:4e:35:fc:fc:d8), Dst: Cisco_f6:1d:ae (00:be:75:f6:1d:ae)
 > Internet Protocol Version 4, Src: 192.168.0.100, Dst: 10.10.1.100
 > Transmission Control Protocol, Src Port: 22195, Dst Port: 80, Seq: 0, Len: 0

Belangrijkste punten:

1. De bron verzendt een TCP/SYN-pakket.
2. Een TCP/RST wordt naar de bron verzonden.
3. De bron geeft de TCP/SYN-pakketten opnieuw door.
4. De MAC-adressen zijn correct (op ingangspakketten behoort het MAC-adres van de bron tot de downstream router, het MAC-adres van de bestemming tot de firewall INSIDE-interface).

Deze afbeelding toont de opname van CAPO in Wireshark:

No.	Time	Source	Destination	Protocol	Length	Info
1	2019-10-11 07:20:36.654507	192.168.0.100	10.10.1.100	TCP	70	22195 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1380
2	2019-10-11 07:20:36.904478	192.168.0.100	10.10.1.100	TCP	70	22196 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1380
3	2019-10-11 07:20:36.904997	10.10.1.100	192.168.0.100	TCP	58	80 → 22196 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
4	2019-10-11 07:20:37.414269	192.168.0.100	10.10.1.100	TCP	70	[TCP Port numbers reused] 22196 → 80 [SYN] Seq=0 Win=0 Len=0
5	2019-10-11 07:20:37.414758	10.10.1.100	192.168.0.100	TCP	58	80 → 22196 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
6	2019-10-11 07:20:37.914305	192.168.0.100	10.10.1.100	TCP	66	[TCP Port numbers reused] 22196 → 80 [SYN] Seq=0 Win=0 Len=0
7	2019-10-11 07:20:37.914762	10.10.1.100	192.168.0.100	TCP	58	80 → 22196 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
8	2019-10-11 07:20:39.654629	192.168.0.100	10.10.1.100	TCP	70	[TCP Retransmission] 22195 → 80 [SYN] Seq=0 Win=0 Len=0
9	2019-10-11 07:20:39.655102	10.10.1.100	192.168.0.100	TCP	58	80 → 22195 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
10	2019-10-11 07:20:40.154700	192.168.0.100	10.10.1.100	TCP	66	[TCP Port numbers reused] 22195 → 80 [SYN] Seq=0 Win=0 Len=0
11	2019-10-11 07:20:40.155173	10.10.1.100	192.168.0.100	TCP	58	80 → 22195 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0

> Frame 1: 70 bytes on wire (560 bits), 70 bytes captured (560 bits)
 > Ethernet II, Src: Cisco_f6:1d:8e (00:be:75:f6:1d:8e), Dst: Cisco_fc:fc:d8 (4c:4e:35:fc:fc:d8)
 > 802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 202
 > Internet Protocol Version 4, Src: 192.168.0.100, Dst: 10.10.1.100
 > Transmission Control Protocol, Src Port: 22195, Dst Port: 80, Seq: 0, Len: 0

Belangrijkste punten:

1. De bron verzendt een TCP/SYN-pakket.
2. Een TCP RST arriveert op de buiteninterface.
3. De bron geeft de TCP/SYN-pakketten opnieuw door.
4. De MAC-adressen zijn correct (op uitgaande pakketten is de firewall buiten de bron-MAC, upstream router is de bestemming-MAC).

Op basis van de 2 captures kan worden geconcludeerd dat:

- De TCP 3-weg handshake tussen de client en de server wordt niet voltooid
- Er is een TCP RST die op de firewall-uitgangsinterface aankomt
- De firewall 'praat' met de juiste upstream en downstream apparaten (gebaseerd op de MAC-adressen)

Aanbevolen acties

De acties die in deze paragraaf worden opgesomd, hebben tot doel de kwestie verder af te zwakken.

Actie 1. Controleer het MAC-adres van de bron dat TCP/RST verstuurt.

Controleer of de doelMAC in het TCP/SYN-pakket hetzelfde is als de bronMAC in het TCP/RST-pakket heeft gezien.

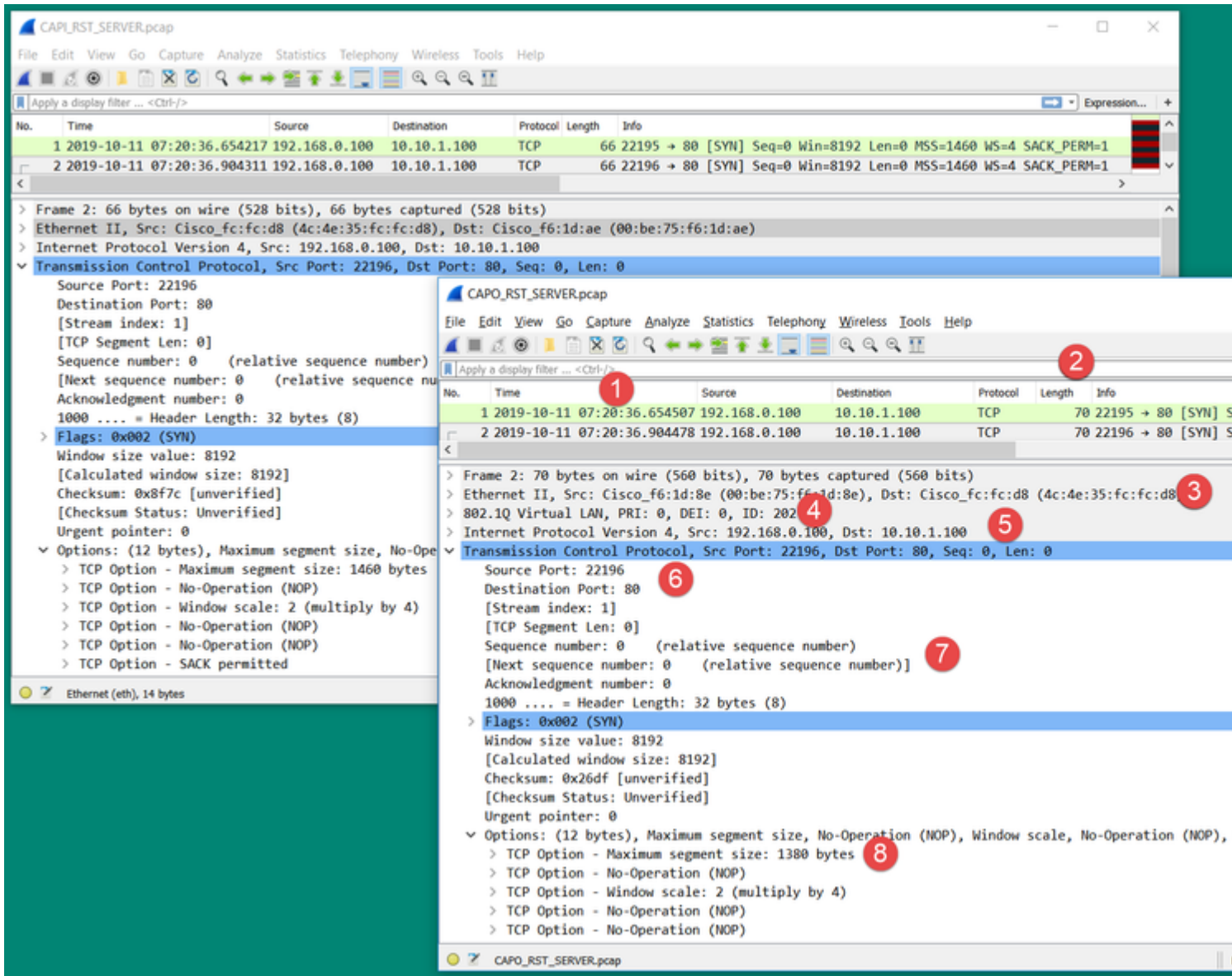
The image displays two screenshots of Wireshark capturing network traffic from a file named 'CAPO_RST_SERVER.pcap'. The top screenshot shows a list of packets where packet 2 is selected. The packet details pane shows an Ethernet II frame with source MAC 'Cisco_f6:1d:8e (00:be:75:f6:1d:8e)' and destination MAC 'Cisco_fc:fc:d8 (4c:4e:35:fc:fc:d8)'. The bottom screenshot shows packet 3 selected, which is a TCP RST, ACK packet from source IP '10.10.1.100' to destination IP '192.168.0.100'. The Ethernet II details show source MAC 'Cisco_fc:fc:d8 (4c:4e:35:fc:fc:d8)' and destination MAC 'Cisco_f6:1d:8e (00:be:75:f6:1d:8e)'. Orange and green boxes highlight these MAC addresses, and arrows indicate the cross-check between the source MAC of the first packet and the destination MAC of the second packet.

Deze check heeft als doel om 2 dingen te bevestigen:

- Controleer of er geen asymmetrische stroom is.
- Controleer dat de MAC tot het verwachte upstream apparaat behoort.

Actie 2. Vergelijk instap- en uitstappakketten.

Vergelijk visueel de 2 pakketten op Wireshark om te verifiëren dat de firewall de pakketten niet aanpast of beschadigt. Enkele verwachte verschillen worden belicht.



Belangrijkste punten:

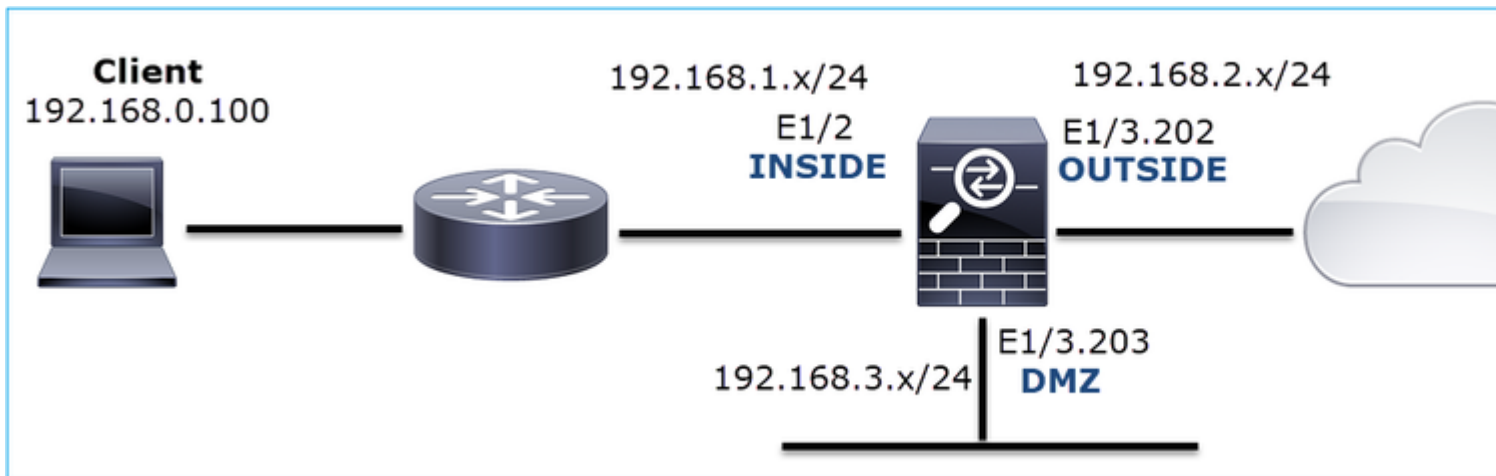
1. Tijdstempels zijn verschillend. Aan de andere kant moet het verschil klein en redelijk zijn. Dit is afhankelijk van de functies en beleidscontroles die op het pakket worden toegepast en van de lading op het apparaat.
2. De lengte van de pakketten kan vooral verschillen als er een dot1Q-header wordt toegevoegd/verwijderd door de firewall aan slechts één kant.
3. De MAC-adressen zijn verschillend.
4. Een dot1Q-header kan aanwezig zijn als de opname op een subinterface werd genomen.
5. Het IP-adres is anders als NAT of PAT (Port Address Translation) wordt toegepast op het pakket.
6. De bron- of doelpoorten zijn verschillend voor het geval dat NAT of PAT op het pakket wordt toegepast.
7. Als u de **optie** Wireshark **Relative Sequence Number** uitschakelt, ziet u dat de TCP-volgnummers/herkenningsnummers door de firewall worden gewijzigd vanwege de randomisatie Initial Sequence Number (ISDN).
8. Sommige TCP-opties kunnen worden overschreven. De firewall verandert bijvoorbeeld standaard de TCP Maximum Segment Size (MSS) in 1380 om pakketfragmentatie in het transportpad te voorkomen.

Actie 3. Neem een opname bij de bestemming.

Indien mogelijk, neem een opname op de bestemming zelf. Als dit niet mogelijk is, neem dan een opname zo dicht mogelijk bij de bestemming. Het doel hier is om te verifiëren wie de TCP RST verstuurt (is de doelserver of is een ander apparaat in het pad?).

Situatie 3. TCP/drieweg-handdruk + RST van één eindpunt

Dit beeld toont de topologie:



Probleembeschrijving: HTTP werkt niet

Beïnvloede stroom:

SRC IP: 192.168.0.10

Laatste IP: 10.10.1.100

Protocol: TCP 80

Capture Analysis

Schakel opnamen in op de FTD LINA engine.

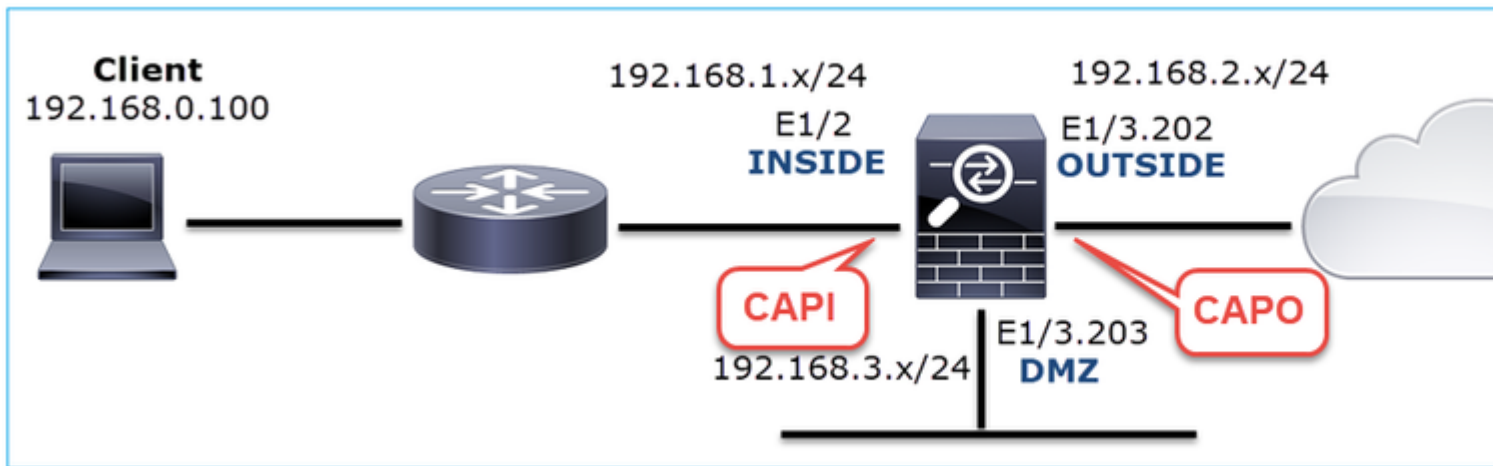
```
<#root>
```

```
firepower#
```

```
capture CAPI int INSIDE match ip host 192.168.0.100 host 10.10.1.100
```

```
firepower#
```

```
capture CAPO int OUTSIDE match ip host 192.168.0.100 host 10.10.1.100
```



Captures - niet-functioneel scenario:

Er zijn een paar verschillende manieren waarop dit probleem zich kan manifesteren in opnamen.

3.1 - TCP 3-weg handdruk + vertraagde RST van de client

Zowel de firewall neemt CAPI op als CAPO bevat dezelfde pakketten, zoals in de afbeelding.

No.	Time	Source	Destination	Protocol	Length	Info
2	2019-10-13 17:06:27.874085	192.168.0.100	10.10.1.100	TCP	66	48295 → 80 [SYN] Seq=179631561 Win=8192 Len=0 MSS=1460 WS=4 SACK_PERM=1
3	2019-10-13 17:06:27.874741	10.10.1.100	192.168.0.100	TCP	66	80 → 48295 [SYN, ACK] Seq=3838911937 Ack=179631562 Win=8192 Len=0 MSS=1380
4	2019-10-13 17:06:27.875183	192.168.0.100	10.10.1.100	TCP	54	48295 → 80 [ACK] Seq=179631562 Ack=3838911938 Win=66240 Len=0
8	2019-10-13 17:06:30.882537	10.10.1.100	192.168.0.100	TCP	66	[TCP Retransmission] 80 → 48295 [SYN, ACK] Seq=3838911937 Ack=179631562 Win=8192 Len=0 MSS=1380
9	2019-10-13 17:06:30.883056	192.168.0.100	10.10.1.100	TCP	66	[TCP Previous segment not captured] 48295 → 80 [ACK] Seq=179631962 Ack=3838911938 Win=66240 Len=0
13	2019-10-13 17:06:36.889022	10.10.1.100	192.168.0.100	TCP	62	[TCP Retransmission] 80 → 48295 [SYN, ACK] Seq=3838911937 Ack=179631562 Win=8192 Len=0 MSS=1380
14	2019-10-13 17:06:36.889526	192.168.0.100	10.10.1.100	TCP	66	[TCP Dup ACK 4#1] 48295 → 80 [ACK] Seq=179631962 Ack=3838911938 Win=66240 Len=0
17	2019-10-13 17:06:47.943631	192.168.0.100	10.10.1.100	TCP	54	48295 → 80 [RST, ACK] Seq=179631962 Ack=3838911938 Win=0 Len=0

Belangrijkste punten:

1. De TCP 3-weg handdruk gaat door de firewall.
2. De server stuurt de SYN/ACK opnieuw door.
3. De client zendt de ACK opnieuw uit.
4. Na ~20 seconden geeft de client het op en verstuurt een TCP RST.

Aanbevolen acties

De acties die in deze paragraaf worden opgesomd, hebben tot doel de kwestie verder af te zwakken.

Actie 1. Neem opnamen zo dicht mogelijk bij de twee eindpunten.

De firewall legt vast dat de client-ACK niet door de server is verwerkt. Dit is gebaseerd op de volgende feiten:

- De server stuurt de SYN/ACK opnieuw door.
- De client zendt de ACK opnieuw uit.
- De client verzendt een TCP/RST of FIN/ACK vóór de gegevens.

Capture op de server toont het probleem. De client ACK van de TCP 3-weg handdruk is nooit angekommen:

26	7.636612	192.168.0.100	10.10.1.100	TCP	66	55324→80 [SYN] Seq=43320132
29	7.637571	10.10.1.100	192.168.0.100	TCP	66	80→55324 [SYN, ACK] Seq=406
30	7.930152	192.168.0.100	10.10.1.100	TCP	66	55325→80 [SYN] Seq=36619749
31	7.930221	10.10.1.100	192.168.0.100	TCP	66	80→55325 [SYN, ACK] Seq=215
41	10.629868	192.168.0.100	10.10.1.100	TCP	66	[TCP Spurious Retransmission]
42	10.633208	10.10.1.100	192.168.0.100	TCP	66	[TCP Retransmission] 80→553
44	10.945178	10.10.1.100	192.168.0.100	TCP	66	[TCP Retransmission] 80→553
60	16.636255	192.168.0.100	10.10.1.100	TCP	62	[TCP Spurious Retransmission]
61	16.639145	10.10.1.100	192.168.0.100	TCP	62	[TCP Retransmission] 80→553
62	16.951195	10.10.1.100	192.168.0.100	TCP	62	[TCP Retransmission] 80→553

3.2 - TCP 3-weg handdruk + vertraagde FIN/ACK van client + vertraagde RST van de server

Zowel de firewall neemt CAPI op als CAPO bevat dezelfde pakketten, zoals in de afbeelding.

25	2019-10-13 17:07:06.853334	192.168.0.100	10.10.1.100	TCP	66	48299 → 80 [SYN] Seq=3239914002 Win=8192 Len=0 MSS=1460 WS=4 S
29	2019-10-13 17:07:09.852922	192.168.0.100	10.10.1.100	TCP	66	[TCP Retransmission] 48299 → 80 [SYN] Seq=3239914002 Win=8192
30	2019-10-13 17:07:09.854844	10.10.1.100	192.168.0.100	TCP	66	80 → 48299 [SYN, ACK] Seq=808763519 Ack=3239914003 Win=8192 L
31	2019-10-13 17:07:09.855287	192.168.0.100	10.10.1.100	TCP	54	48299 → 80 [ACK] Seq=3239914003 Ack=808763520 Win=66240 Len=0
34	2019-10-13 17:07:14.856996	192.168.0.100	10.10.1.100	TCP	54	48299 → 80 [FIN, ACK] Seq=3239914003 Ack=808763520 Win=66240 L
35	2019-10-13 17:07:15.861451	10.10.1.100	192.168.0.100	TCP	62	[TCP Retransmission] 80 → 48299 [SYN, ACK] Seq=808763519 Ack=3
36	2019-10-13 17:07:15.861970	192.168.0.100	10.10.1.100	TCP	66	[TCP Dup ACK 31#1] 48299 → 80 [ACK] Seq=3239914004 Ack=8087635
39	2019-10-13 17:07:17.854051	192.168.0.100	10.10.1.100	TCP	54	[TCP Retransmission] 48299 → 80 [FIN, ACK] Seq=3239914003 Ack=
40	2019-10-13 17:07:23.855012	192.168.0.100	10.10.1.100	TCP	54	[TCP Retransmission] 48299 → 80 [FIN, ACK] Seq=3239914003 Ack=
46	2019-10-13 17:07:27.858949	10.10.1.100	192.168.0.100	TCP	54	80 → 48299 [RST] Seq=808763520 Win=0 Len=0

Belangrijkste punten:

1. De TCP 3-weg handdruk gaat door de firewall.
2. Na 5 seconden stuurt de client een FIN/ACK.
3. Na ~20 seconden geeft de server het op en verstuurt een TCP RST.

Gebaseerd op deze opname kan worden geconcludeerd dat hoewel er een TCP 3-weg handdruk door de firewall is het lijkt dat het nooit echt wordt voltooid op één eindpunt (de heruitzendingen wijzen dit).

Aanbevolen acties

Hetzelfde als in geval 3.1

3.3 - TCP 3-weg handdruk + vertraagde RST van de client

Zowel de firewall neemt CAPI op als CAPO bevat dezelfde pakketten, zoals in de afbeelding.

No.	Time	Source	Destination	Protocol	Length	Info
129	2019-10-13 17:09:20.513355	192.168.0.100	10.10.1.100	TCP	66	48355 → 80 [SYN] Seq=2581697538 Wi
130	2019-10-13 17:09:20.514011	10.10.1.100	192.168.0.100	TCP	66	80 → 48355 [SYN, ACK] Seq=16330186
131	2019-10-13 17:09:20.514438	192.168.0.100	10.10.1.100	TCP	54	48355 → 80 [ACK] Seq=2581697539 Ac
132	2019-10-13 17:09:39.473089	192.168.0.100	10.10.1.100	TCP	54	48355 → 80 [RST, ACK] Seq=25816979

Belangrijkste punten:

1. De TCP 3-weg handdruk gaat door de firewall.
2. Na ~20 seconden geeft de client het op en verstuurt een TCP RST.

Op basis van deze gegevens kan worden geconcludeerd dat:

- Na 5-20 seconden geeft een eindpunt op en besluit de verbinding te beëindigen.

Aanbevolen acties

Hetzelfde als in geval 3.1

3.4 - TCP 3-weg handdruk + directe RST van de server

Beide firewalls nemen CAPI en CAPO bevatten deze pakketten, zoals in de afbeelding.

No.	Time	Source	Destination	Protocol	Length	Info
26	2019-10-13 17:07:07.104410	192.168.0.100	10.10.1.100	TCP	66	48300 → 80 [SYN] Seq=2563435279 Win=
27	2019-10-13 17:07:07.105112	10.10.1.100	192.168.0.100	TCP	66	80 → 48300 [SYN, ACK] Seq=3757137497
28	2019-10-13 17:07:07.105554	192.168.0.100	10.10.1.100	TCP	54	48300 → 80 [ACK] Seq=2563435280 Ack=
41	2019-10-13 17:07:07.106325	10.10.1.100	192.168.0.100	TCP	54	80 → 48300 [RST] Seq=2563435280 Win=

Belangrijkste punten:

1. De TCP 3-weg handdruk gaat door de firewall.
2. Er is een TCP RST van de server een paar milliseconden na het ACK pakket.

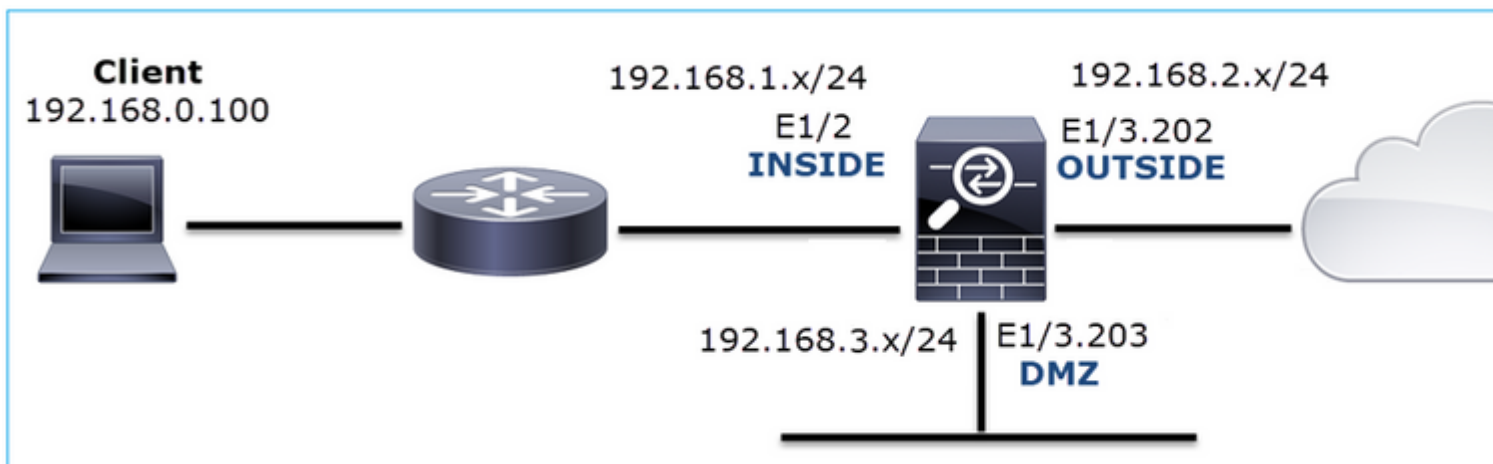
Aanbevolen acties

Actie: Neem opnamen zo dicht mogelijk bij de server.

Een directe TCP RST van de server kan wijzen op een defecte server of een apparaat in het pad dat de TCP RST verstuurt. Neem een opname op de server zelf en bepaal de bron van TCP/RST.

Situatie 4. TCP/RST vanaf de client

Dit beeld toont de topologie:



Probleembeschrijving: HTTP werkt niet.

Beïnvloede stroom:

SRC IP: 192.168.0.10

Laatste IP: 10.10.1.100

Protocol: TCP 80

Capture Analysis

Schakel opnamen in op de FTD LINA-motor.

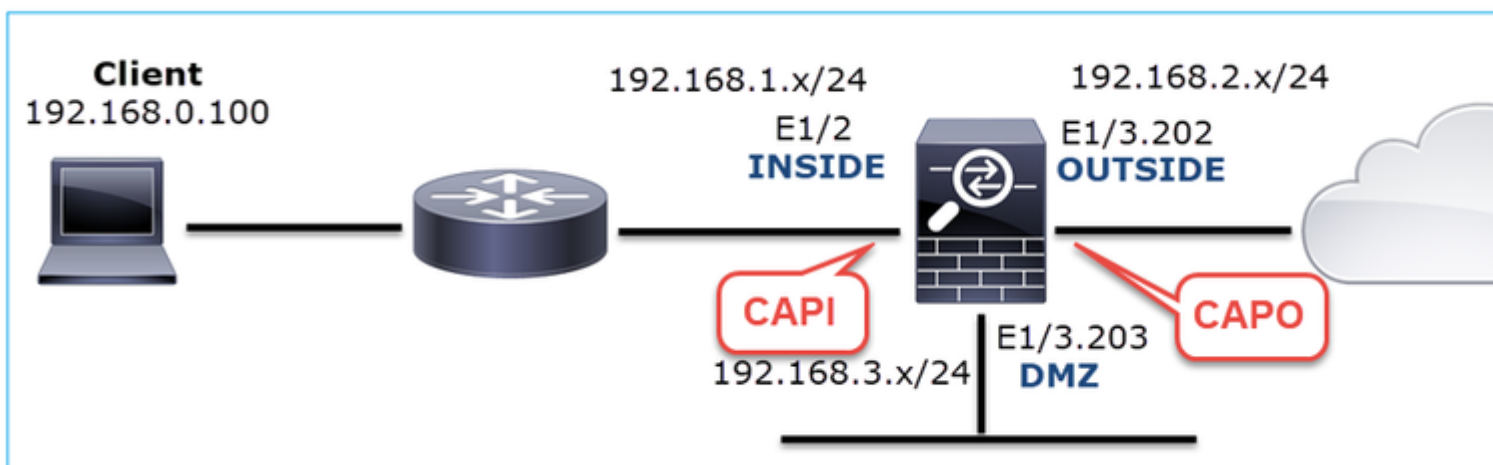
```
<#root>
```

```
firepower#
```

```
capture CAPI int INSIDE match ip host 192.168.0.100 host 10.10.1.100
```

```
firepower#
```

```
capture CAPO int OUTSIDE match ip host 192.168.0.100 host 10.10.1.100
```



Captures - niet-functioneel scenario:

Dit zijn de CAPI-inhoud.

```
<#root>
```

```
firepower#
```

```
show capture CAPI
```

14 packets captured

```
1: 12:32:22.860627 192.168.0.100.47078 > 10.10.1.100.80: S 4098574664:4098574664(0) win 8192 <mss 1
2: 12:32:23.111307 192.168.0.100.47079 > 10.10.1.100.80: S 2486945841:2486945841(0) win 8192 <mss 1
3: 12:32:23.112390 192.168.0.100.47079 > 10.10.1.100.80: R 3000518858:3000518858(0) win 0
4: 12:32:25.858109 192.168.0.100.47078 > 10.10.1.100.80: S 4098574664:4098574664(0) win 8192 <mss 1
5: 12:32:25.868698 192.168.0.100.47078 > 10.10.1.100.80: R 1386249853:1386249853(0) win 0
```

```
6: 12:32:26.108118 192.168.0.100.47079 > 10.10.1.100.80: S 2486945841:2486945841(0) win 8192 <mss 1
7: 12:32:26.109079 192.168.0.100.47079 > 10.10.1.100.80: R 3000518858:3000518858(0) win 0
8: 12:32:26.118295 192.168.0.100.47079 > 10.10.1.100.80: R 3000518858:3000518858(0) win 0
9: 12:32:31.859925 192.168.0.100.47078 > 10.10.1.100.80: S 4098574664:4098574664(0) win 8192 <mss 1
10: 12:32:31.860902 192.168.0.100.47078 > 10.10.1.100.80: R 1386249853:1386249853(0) win 0
11: 12:32:31.875229 192.168.0.100.47078 > 10.10.1.100.80: R 1386249853:1386249853(0) win 0
12: 12:32:32.140632 192.168.0.100.47079 > 10.10.1.100.80: R 3000518858:3000518858(0) win 0
13: 12:32:32.159995 192.168.0.100.47079 > 10.10.1.100.80: S 2486945841:2486945841(0) win 8192 <mss 1
14: 12:32:32.160956 192.168.0.100.47079 > 10.10.1.100.80: R 3000518858:3000518858(0) win 0
14 packets shown
```

Dit zijn de CAPO inhoud:

```
<#root>
```

```
firepower#
```

```
show capture CAPO
```

11 packets captured

```
1: 12:32:22.860780 802.1Q vlan#202 P0 192.168.0.100.47078 > 10.10.1.100.80: S 1386249852:1386249852
2: 12:32:23.111429 802.1Q vlan#202 P0 192.168.0.100.47079 > 10.10.1.100.80: S 3000518857:3000518857
3: 12:32:23.112405 802.1Q vlan#202 P0 192.168.0.100.47079 > 10.10.1.100.80: R 3514091874:3514091874
4: 12:32:25.858125 802.1Q vlan#202 P0 192.168.0.100.47078 > 10.10.1.100.80: S 1386249852:1386249852
5: 12:32:25.868729 802.1Q vlan#202 P0 192.168.0.100.47078 > 10.10.1.100.80: R 2968892337:2968892337
6: 12:32:26.108240 802.1Q vlan#202 P0 192.168.0.100.47079 > 10.10.1.100.80: S 3822259745:3822259745
7: 12:32:26.109094 802.1Q vlan#202 P0 192.168.0.100.47079 > 10.10.1.100.80: R 40865466:40865466(0)
8: 12:32:31.860062 802.1Q vlan#202 P0 192.168.0.100.47078 > 10.10.1.100.80: S 4294058752:4294058752
9: 12:32:31.860917 802.1Q vlan#202 P0 192.168.0.100.47078 > 10.10.1.100.80: R 1581733941:1581733941
10: 12:32:32.160102 802.1Q vlan#202 P0 192.168.0.100.47079 > 10.10.1.100.80: S 4284301197:4284301197
11: 12:32:32.160971 802.1Q vlan#202 P0 192.168.0.100.47079 > 10.10.1.100.80: R 502906918:502906918(0)
11 packets shown
```

De firewalllogboeken tonen:

```
<#root>
```

```
firepower#
```

```
show log | i 47741
```

```
Oct 13 2019 13:57:36: %FTD-6-302013: Built inbound TCP connection 4869 for INSIDE:192.168.0.100/47741 (1
Oct 13 2019 13:57:36: %FTD-6-302014: Teardown TCP connection 4869 for INSIDE:192.168.0.100/47741 to OUTS
```

```
TCP Reset-O from INSIDE
```

```
Oct 13 2019 13:57:39: %FTD-6-302013: Built inbound TCP connection 4870 for INSIDE:192.168.0.100/47741 (1
Oct 13 2019 13:57:39: %FTD-6-302014: Teardown TCP connection 4870 for INSIDE:192.168.0.100/47741 to OUTS
```

```
TCP Reset-O from INSIDE
```

```
Oct 13 2019 13:57:45: %FTD-6-302013: Built inbound TCP connection 4871 for INSIDE:192.168.0.100/47741 (1
Oct 13 2019 13:57:45: %FTD-6-302014: Teardown TCP connection 4871 for INSIDE:192.168.0.100/47741 to OUTS
```

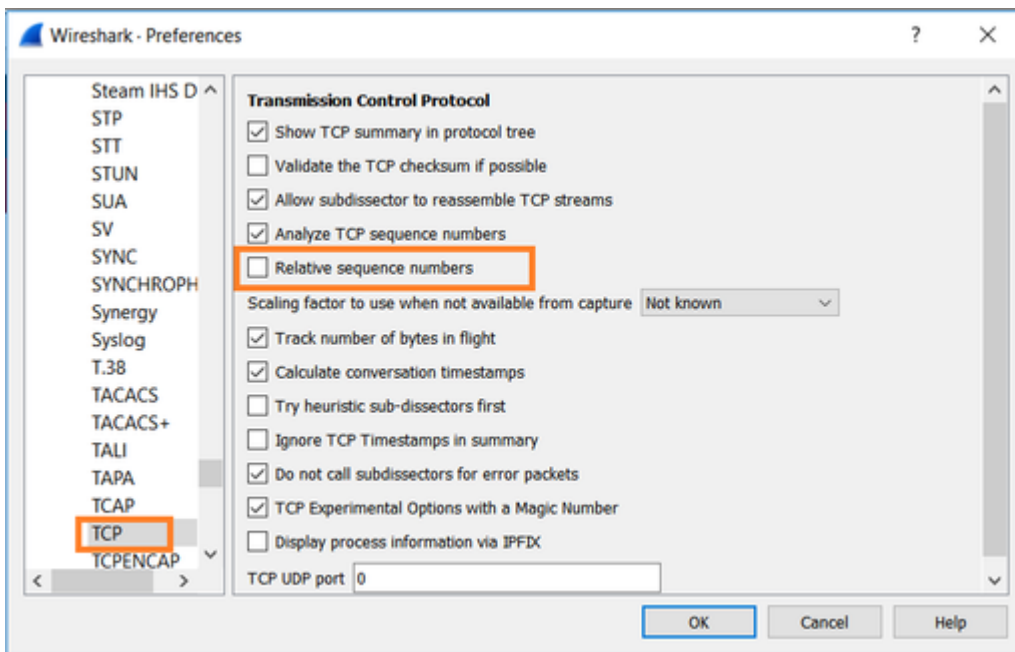
Deze logboeken geven aan dat er een TCP RST is die op firewall INSIDE interface aankomt

CAPI Capture in Wireshark:

Volg de eerste TCP-stream, zoals in de afbeelding.

No.	Time	Source	Destination	Protocol	Length	Info
1	2019-10-13 14:32:22.860627	192.168.0.100	10.10.1.100	TCP	66	47078 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=4 SACK_
2	2019-10-13 14:32:23.111307	192.168.0.100	10.10.1.100	TCP	66	47079 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=4 SACK_
3	2019-10-13 14:32:23.112390	192.168.0.100	10.10.1.100	TCP	54	47079 → 80 [RST] Seq=513573017 Win=0 Len=0
4	2019-10-13 14:32:25.858109	192.168.0.100	10.10.1.100	TCP	66	[TCP Retransmission] 47078 → 80 [SYN] Seq=0 Win=8192 Len=
5	2019-10-13 14:32:25.868698	192.168.0.100	10.10.1.100	TCP	54	47078 → 80 [RST] Seq=1582642485 Win=0 Len=0
6	2019-10-13 14:32:26.108118	192.168.0.100	10.10.1.100	TCP	66	[TCP Retransmission] 47079 → 80 [SYN] Seq=0 Win=8192 Len=
7	2019-10-13 14:32:26.109079	192.168.0.100	10.10.1.100	TCP	54	47079 → 80 [RST] Seq=513573017 Win=0 Len=0
8	2019-10-13 14:32:26.118295	192.168.0.100	10.10.1.100	TCP	54	47079 → 80 [RST] Seq=513573017 Win=0 Len=0
9	2019-10-13 14:32:31.859925	192.168.0.100	10.10.1.100	TCP	62	[TCP Retransmission] 47078 → 80 [SYN] Seq=0 Win=8192 Len=
10	2019-10-13 14:32:31.860902	192.168.0.100	10.10.1.100	TCP	54	47078 → 80 [RST] Seq=1582642485 Win=0 Len=0
11	2019-10-13 14:32:31.875229	192.168.0.100	10.10.1.100	TCP	54	47078 → 80 [RST] Seq=1582642485 Win=0 Len=0
12	2019-10-13 14:32:32.140632	192.168.0.100	10.10.1.100	TCP	54	47079 → 80 [RST] Seq=513573017 Win=0 Len=0
13	2019-10-13 14:32:32.159995	192.168.0.100	10.10.1.100	TCP	62	[TCP Retransmission] 47079 → 80 [SYN] Seq=0 Win=8192 Len=
14	2019-10-13 14:32:32.160956	192.168.0.100	10.10.1.100	TCP	54	47079 → 80 [RST] Seq=513573017 Win=0 Len=0

Navigeer onder **Wireshark** naar **Bewerken > Voorkeuren > Protocollen > TCP** en hef de selectie van de optie **Relatieve volgnummers** zoals in de afbeelding op.



Dit beeld toont de inhoud van de eerste stroom in CAPI Capi Capture:

No.	Time	Source	Destination	Protocol	Length	Info
1	2019-10-13 14:32:22.860627	192.168.0.100	10.10.1.100	TCP	66	47078 → 80 [SYN] Seq=4098574664 Win=8192 Len=0 MSS=1460 WS=4 SACK_PERM=1
4	2019-10-13 14:32:25.858109	192.168.0.100	10.10.1.100	TCP	66	[TCP Retransmission] 47078 → 80 [SYN] Seq=4098574664 Win=8192 Len=0 MSS=1460 WS=4 SACK_PERM=1
5	2019-10-13 14:32:25.868698	192.168.0.100	10.10.1.100	TCP	54	47078 → 80 [RST] Seq=1386249853 Win=0 Len=0
9	2019-10-13 14:32:31.859925	192.168.0.100	10.10.1.100	TCP	62	[TCP Retransmission] 47078 → 80 [SYN] Seq=4098574664 Win=8192 Len=0 MSS=1460 WS=4 SACK_PERM=1
10	2019-10-13 14:32:31.860902	192.168.0.100	10.10.1.100	TCP	54	47078 → 80 [RST] Seq=1386249853 Win=0 Len=0
11	2019-10-13 14:32:31.875229	192.168.0.100	10.10.1.100	TCP	54	47078 → 80 [RST] Seq=1386249853 Win=0 Len=0


```

> Frame 1: 66 bytes on wire (528 bits), 66 bytes captured (528 bits)
> Ethernet II, Src: Cisco_fc:fc:d8 (4c:4e:35:fc:fc:d8), Dst: Cisco_f6:1d:ae (00:be:75:f6:1d:ae)
> Internet Protocol Version 4, Src: 192.168.0.100, Dst: 10.10.1.100
> Transmission Control Protocol, Src Port: 47078, Dst Port: 80, Seq: 4098574664, Len: 0
  Source Port: 47078
  Destination Port: 80
  [Stream index: 0]
  [TCP Segment Len: 0]
  Sequence number: 4098574664
  [Next sequence number: 4098574664]
  Acknowledgment number: 0
  1000 .... = Header Length: 32 bytes (8)
  > Flags: 0x002 (SYN)
  Window size value: 8192
  [Calculated window size: 8192]
  Checksum: 0x8cd1 [unverified]
  [Checksum Status: Unverified]
  Urgent pointer: 0
  > Options: (12 bytes), Maximum segment size, No-Operation (NOP), Window scale, No-Operation (NOP), No-Operation (NOP), SACK permitted
  > [Timestamps]

```

Belangrijkste punten:

1. De client verzendt een TCP/SYN-pakket.
2. De client verzendt een TCP/RST-pakket.
3. Het TCP/SYN-pakket heeft een waarde voor het volgnummer gelijk aan 4098574664.

De zelfde stroom in CAPO-opname bevat:

No.	Time	Source	Destination	Protocol	Length	Info
1	2019-10-13 14:32:22.860780	192.168.0.100	10.10.1.100	TCP	70	47078 → 80 [SYN] Seq=1386249852 Win=8192 Len=0 MSS=1460 WS=4 SACK_PERM=1
4	2019-10-13 14:32:25.858125	192.168.0.100	10.10.1.100	TCP	70	[TCP Retransmission] 47078 → 80 [SYN] Seq=1386249852 Win=8192 Len=0 MSS=1460 WS=4 SACK_PERM=1
5	2019-10-13 14:32:25.868729	192.168.0.100	10.10.1.100	TCP	58	47078 → 80 [RST] Seq=2968892337 Win=0 Len=0


```

> Frame 1: 70 bytes on wire (560 bits), 70 bytes captured (560 bits)
> Ethernet II, Src: Cisco_f6:1d:8e (00:be:75:f6:1d:8e), Dst: Cisco_fc:fc:d8 (4c:4e:35:fc:fc:d8)
> 802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 202
> Internet Protocol Version 4, Src: 192.168.0.100, Dst: 10.10.1.100
> Transmission Control Protocol, Src Port: 47078, Dst Port: 80, Seq: 1386249852, Len: 0

```

Belangrijkste punten:

1. De client verzendt een TCP/SYN-pakket. De firewall randomiseert ISDN.
2. De client verzendt een TCP/RST-pakket.

Op basis van de twee opnamen kan worden geconcludeerd dat:

- Er is geen TCP 3-weg handdruk tussen de client en de server.
- Er is een TCP RST die van de client komt. De waarde van het TCP/RST-volgnummer in CAPI-opname is 1386249853.

Aanbevolen acties

De acties die in deze paragraaf worden opgesomd, hebben tot doel de kwestie verder af te zwakken.

Actie 1. Neem een opname op de client.

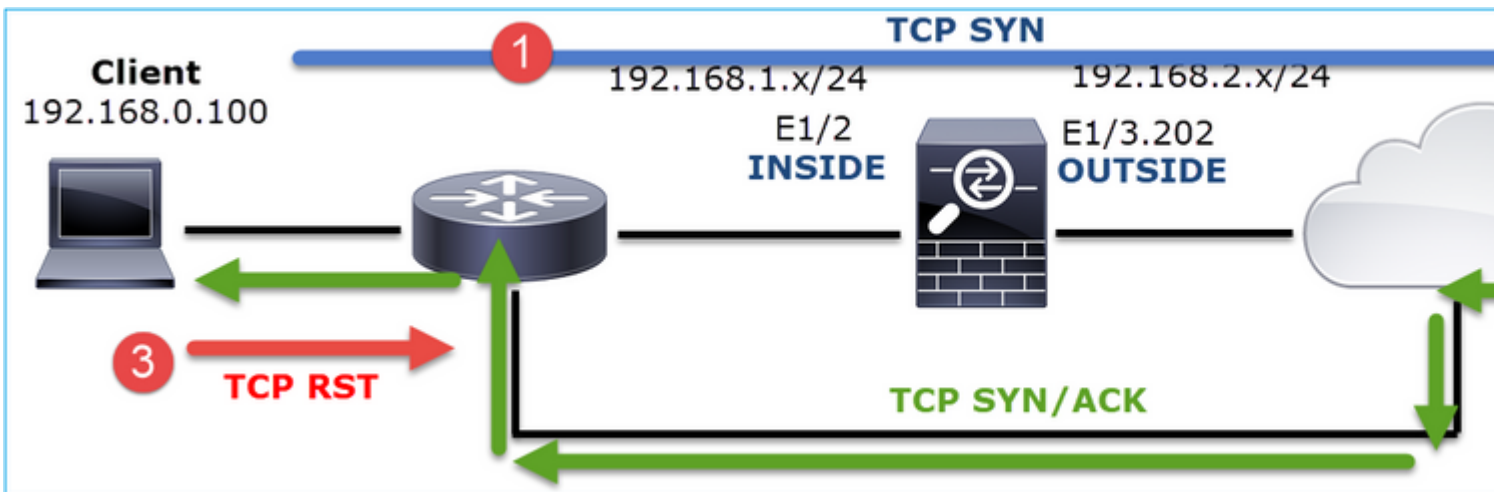
Op basis van de opnamen die zijn verzameld op de firewall is er een sterke aanwijzing voor een asymmetrische stroom. Dit is gebaseerd op het feit dat de client een TCP RST met een waarde van 1386249853 (het gerandomiseerde ISDN) verstuurt:

No.	Time	Source	Destination	Protocol	Length	Info
19	6.040337	192.168.0.100	10.10.1.100	TCP	66	47078→80 [SYN] Seq=4098574664
29	9.037499	192.168.0.100	10.10.1.100	TCP	66	[TCP Retransmission] 47078→80 [SYN] Seq=4098574664
30	9.048155	10.10.1.100	192.168.0.100	TCP	66	[TCP ACKed unseen segment] 80→47078 [SYN, A
31	9.048184	192.168.0.100	10.10.1.100	TCP	54	47078→80 [RST] Seq=1386249853 Win=0 Len=0

Belangrijkste punten:

1. De client verzendt een TCP/SYN-pakket. Het volgnummer is 4098574664 en is hetzelfde als het nummer dat wordt weergegeven op de firewall INSIDE interface (CAPI)
2. Er is een TCP SYN/ACK met ACK nummer 1386249853 (wat verwacht wordt vanwege ISDN randomisering). Dit pakket is niet weergegeven in de firewallopnamen
3. De client stuurt een TCP/RST omdat er een SYN/ACK met ACK-nummerwaarde van 4098574665 verwacht werd, maar het ontving 1386249853 waarde

Dit kan als volgt worden weergegeven:

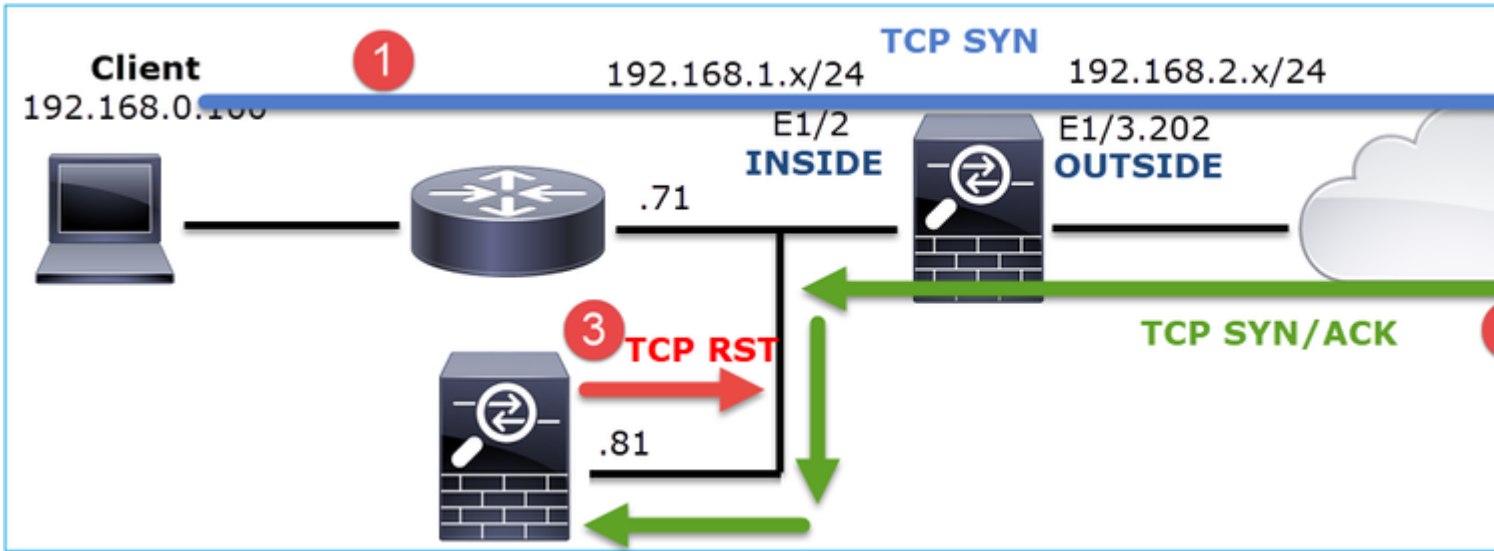


Actie 2. Controleer de routing tussen de client en de firewall.

Bevestig dat:

- De MAC-adressen in de opnamen zijn de verwachte adressen.
- Zorg ervoor dat de routing tussen de firewall en de client symmetrisch is.

Er zijn scenario's waar RST uit een apparaat komt dat tussen de firewall en de cliënt zit terwijl er een asymmetrische routing in het interne netwerk is. In het beeld wordt een typisch geval weergegeven:



In dit geval heeft de opname deze inhoud. Let op het verschil tussen het MAC-adres van de bron van het TCP/SYN-pakket en het MAC-adres van de bron van het TCP/RST en het MAC-adres van de bestemming van het TCP/SYN/ACK-pakket:

```
<#root>
firepower#
show capture CAPI detail
  1: 13:57:36.730217
  4c4e.35fc.fcd8
  00be.75f6.1dae 0x0800 Length: 66
    192.168.0.100.47740 > 10.10.1.100.80: S [tcp sum ok] 3045001876:3045001876(0) win 8192 <mss 1460,r
  2: 13:57:36.981104 4c4e.35fc.fcd8 00be.75f6.1dae 0x0800 Length: 66
    192.168.0.100.47741 > 10.10.1.100.80: S [tcp sum ok] 3809380540:3809380540(0) win 8192 <mss 1460,r
  3: 13:57:36.981776 00be.75f6.1dae
a023.9f92.2a4d
  0x0800 Length: 66
    10.10.1.100.80 > 192.168.0.100.47741: S [tcp sum ok] 1304153587:1304153587(0) ack 3809380541 win 8
  4: 13:57:36.982126
a023.9f92.2a4d
  00be.75f6.1dae 0x0800 Length: 54
    192.168.0.100.47741 > 10.10.1.100.80:
R
[ tcp sum ok] 3809380541:3809380541(0) ack 1304153588 win 8192 (ttl 255, id 48501)
...
```

Situatie 5. Langzame TCP-overdracht (scenario 1)

Probleembeschrijving:

SFTP-overdracht tussen hosts 10.11.4.171 en 10.77.19.11 verloopt traag. Hoewel de minimale bandbreedte (BW) tussen de 2 hosts 100 Mbps is, gaat de overdrachtsnelheid niet verder dan 5 Mbps.

Tegelijkertijd is de overdrachtssnelheid tussen gastheren 10.11.2.124 en 172.25.18.134 vrij hoger.

Achtergrondinformatie:

De maximale overdrachtsnelheid voor één TCP-stroom wordt bepaald door het Bandwidth Delay Product (BDP). De gebruikte formule is in de afbeelding weergegeven:

$$\text{Max Single TCP Flow Throughput [bps]} = \frac{\text{TCP Window (Bytes)}}{\text{RTT (Seconds)}} \times 8 \text{ [bits/Byte]}$$

Voor meer informatie over de BDP check je hier de resources:

- [Waarom gebruikt uw toepassing alleen 10Mbps en is de link 1 Gbps?](#)
- [BRKSEC-3021 - Geavanceerd - Firewallprestaties maximaliseren](#)

Scenario 1. Langzame overdracht

Dit beeld toont de topologie:



Beïnvloede stroom:

SRC IP: 10.11.4.171

Laatste IP: 10.7.19.11

Protocol: SFTP (FTP over SSH)

Capture Analysis

Opnamen op FTD LINA-motor inschakelen:

<#root>

```
firepower#
```

```
capture CAPI int INSIDE buffer 33554432 match ip host 10.11.4.171 host 10.77.19.11
```

```
firepower#
```

```
capture CAPO int OUTSIDE buffer 33554432 match ip host 10.11.4.171 host 10.77.19.11
```

Waarschuwing: LINA legt vast op FP1xxx en FP21xx Captures beïnvloeden de overdrachtsnelheid van verkeer dat door de FTD gaat. Schakel LINA niet in wanneer u problemen oplost met de prestaties (langzame overdracht via de FTD) op FP1xxx- en FP21xxx-platforms. Gebruik in plaats daarvan SPAN of een HW Tap-apparaat naast de opnamen op de bron- en doelhosts. Het probleem is gedocumenteerd in Cisco bug-id [CSCvo30697](#).

```
<#root>
```

```
firepower#
```

```
capture CAPI type raw-data trace interface inside match icmp any any
```

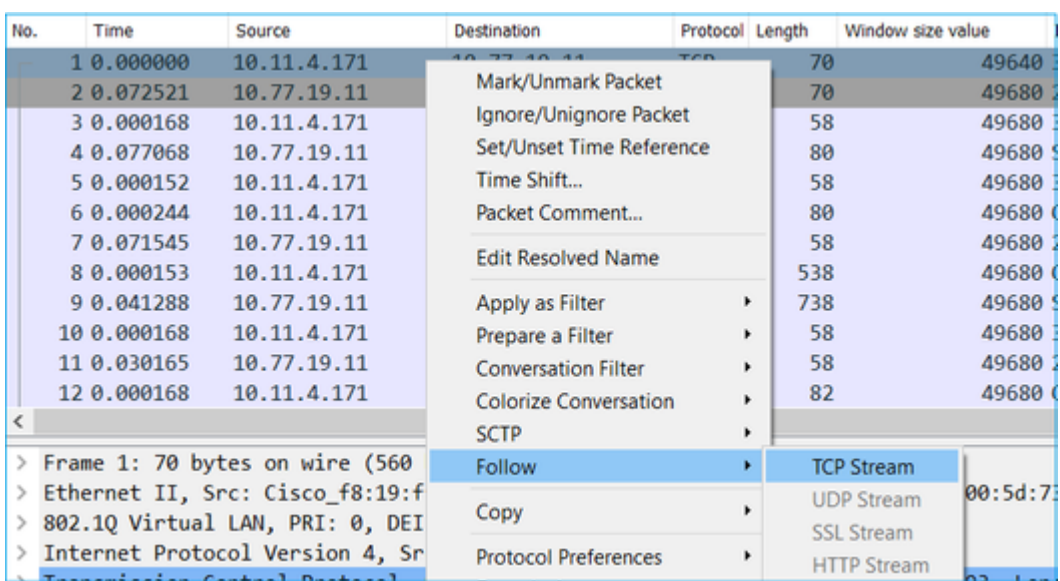
```
WARNING: Running packet capture can have an adverse impact on performance.
```

Aanbevolen acties

De acties die in deze paragraaf worden opgesomd, hebben tot doel de kwestie verder af te zwakken.

Ronde reistijd (RTT)

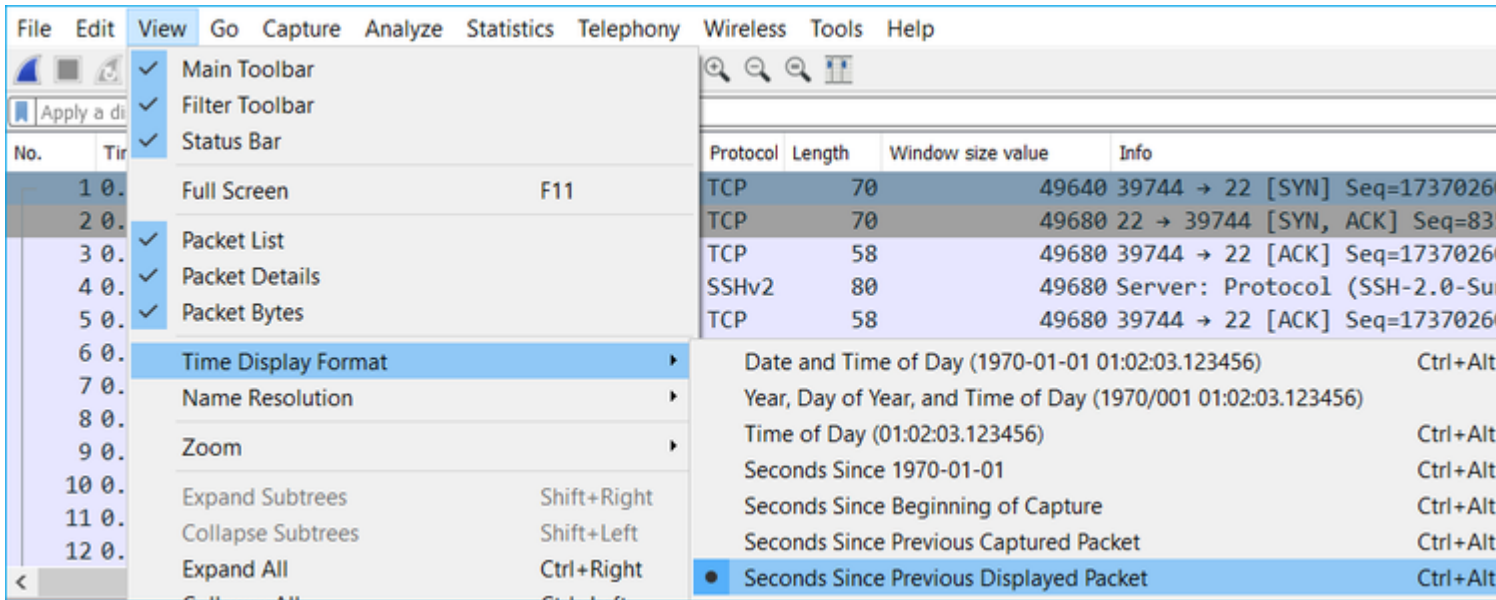
Identificeer eerst de overdrachtstroom en volg deze:



No.	Time	Source	Destination	Protocol	Length	Window size value
1	0.000000	10.11.4.171	10.77.19.11	TCP	70	49640
2	0.072521	10.77.19.11	10.11.4.171	TCP	70	49680
3	0.000168	10.11.4.171	10.77.19.11	TCP	58	49680
4	0.077068	10.77.19.11	10.11.4.171	TCP	80	49680
5	0.000152	10.11.4.171	10.77.19.11	TCP	58	49680
6	0.000244	10.11.4.171	10.77.19.11	TCP	80	49680
7	0.071545	10.77.19.11	10.11.4.171	TCP	58	49680
8	0.000153	10.11.4.171	10.77.19.11	TCP	538	49680
9	0.041288	10.77.19.11	10.11.4.171	TCP	738	49680
10	0.000168	10.11.4.171	10.77.19.11	TCP	58	49680
11	0.030165	10.77.19.11	10.11.4.171	TCP	58	49680
12	0.000168	10.11.4.171	10.77.19.11	TCP	82	49680

The screenshot shows a context menu for a selected packet (No. 1) with the following options: Mark/Unmark Packet, Ignore/Unignore Packet, Set/Unset Time Reference, Time Shift..., Packet Comment..., Edit Resolved Name, Apply as Filter, Prepare a Filter, Conversation Filter, Colorize Conversation, Sctp, Follow (selected), Copy, and Protocol Preferences. The 'Follow' option is expanded to show 'TCP Stream', 'UDP Stream', 'SSL Stream', and 'HTTP Stream'. The 'TCP Stream' option is highlighted.

Verander de weergave Wireshark om de **seconden sinds het vorige weergegeven pakket** weer te geven. Dit vergemakkelijkt de berekening van de RTT:



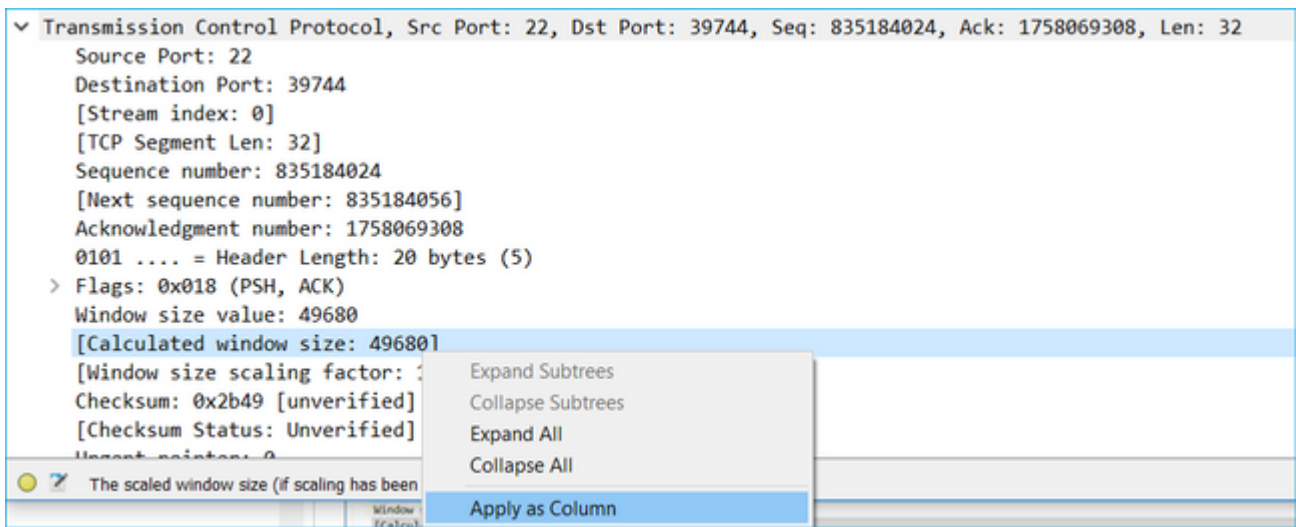
De RTT kan worden berekend door de tijdswaarden tussen twee pakketuitwisselingen op te tellen (één naar de bron en één naar de bestemming). In dit geval toont #2 de RTT tussen de firewall en het apparaat dat het SYN/ACK-pakket (server) heeft verzonden. Packet #3 toont de RTT tussen de firewall en het apparaat dat het ACK-pakket (client) heeft verzonden. De toevoeging van de 2 getallen levert een goede schatting van de end-to-end RTT:

No.	Time	Source	Destination	Protocol	Length	Window size value	Info
1	0.000000	10.11.4.171	10.77.19.11	TCP	70	49640 39744 → 22	[SYN] Seq=1737026093 Win=49640 Len=0
2	0.072521	10.77.19.11	10.11.4.171	TCP	70	49680 22 → 39744	[SYN, ACK] Seq=835172681 Ack=1737026093 Win=0 Len=0
3	0.000168	10.11.4.171	10.77.19.11	TCP	58	49680 39744 → 22	[ACK] Seq=1737026094 Ack=835172682 Win=0 Len=0
4	0.077068	10.77.19.11	10.11.4.171	SSHv2	80		49680 Server: Protocol (SSH-2.0-Sun_SSH_1.1.8)
5	0.000152	10.11.4.171	10.77.19.11	TCP	58	49680 39744 → 22	[ACK] Seq=1737026094 Ack=835172704 Win=0 Len=0
6	0.000244	10.11.4.171	10.77.19.11	SSHv2	80		49680 Client: Protocol (SSH-2.0-Sun_SSH_1.1.4)
7	0.071545	10.77.19.11	10.11.4.171	TCP	58	49680 22 → 39744	[ACK] Seq=835172704 Ack=1737026116 Win=0 Len=0
8	0.000153	10.11.4.171	10.77.19.11	SSHv2	538		49680 Client: Key Exchange Init
9	0.041288	10.77.19.11	10.11.4.171	SSHv2	738		49680 Server: Key Exchange Init
10	0.000168	10.11.4.171	10.77.19.11	TCP	58	49680 39744 → 22	[ACK] Seq=1737026596 Ack=835173384 Win=0 Len=0
11	0.030165	10.77.19.11	10.11.4.171	TCP	58	49680 22 → 39744	[ACK] Seq=835173384 Ack=1737026596 Win=0 Len=0
12	0.000168	10.11.4.171	10.77.19.11	SSHv2	82		49680 Client: Diffie-Hellman Group Exchange Request

RTT ≈ 80 msec

Berekening van TCP-venstergrootte

Breid een TCP-pakket uit, vouw de TCP-header uit, selecteer **Berekende venstergrootte** en selecteer **Toepassen als kolom**:



Controleer in de kolom **Berekende venstergrootte** wat de maximale venstergrootte is tijdens de TCP-sessie. U kunt ook de waarden op de kolomnaam selecteren en sorteren.

Als u een bestand downloadt (**server > client**), moet u de waarden controleren die op de server worden geadverteerd. De maximale venstergrootte die door de server wordt geadverteerd bepaalt de maximale overdrachtsnelheid.

In dit geval is de grootte van het TCP-venster $\hat{=} 50000$ bytes

No.	Time	Source	Destination	Protocol	Length	Calculated window size	Info
24...	0.000091	10.11.4.171	10.77.19.11	TCP	58	49680	39744 → 22 [ACK] Seq=1758069341
24...	0.000077	10.77.19.11	10.11.4.171	TCP	58	49680	22 → 39744 [FIN, ACK] Seq=835184024
24...	0.071605	10.77.19.11	10.11.4.171	TCP	58	49680	22 → 39744 [ACK] Seq=835184152
24...	0.000153	10.11.4.171	10.77.19.11	TCP	58	49680	39744 → 22 [FIN, ACK] Seq=1758069308
24...	0.000443	10.11.4.171	10.77.19.11	SSHv2	90	49680	Client: Encrypted packet (len=32)
24...	0.071666	10.77.19.11	10.11.4.171	SSHv2	154	49680	Server: Encrypted packet (len=96)
24...	0.044050	10.11.4.171	10.77.19.11	TCP	58	49680	39744 → 22 [ACK] Seq=1758069308
24...	0.073605	10.77.19.11	10.11.4.171	SSHv2	90	49680	Server: Encrypted packet (len=32)
24...	0.000747	10.11.4.171	10.77.19.11	SSHv2	90	49680	Client: Encrypted packet (len=32)

Gebaseerd op deze waarden en met behulp van de Bandwidth Delay Product formule krijgt u de maximale theoretische bandbreedte die onder deze omstandigheden kan worden bereikt: $50000 * 8 / 0.08 = 5$ Mbps maximale theoretische bandbreedte.

Dit komt overeen met wat de klant in dit geval ervaart.

Controleer de TCP 3-voudige handdruk nauwkeurig. Beide kanten, en nog belangrijker de server, adverteren een vensterschaalwaarde van 0 wat $2^0 = 1$ betekent (geen vensterschaling). Dit heeft een negatieve invloed op de overdrachtssnelheid:

No.	Time	Source	Destination	Protocol	Length	Window size value	Info
1	0.000000	10.11.4.171	10.77.19.11	TCP	70	49640 39744 → 22 [SYN] Seq=1737026093 Win=49640 L	
2	0.072521	10.77.19.11	10.11.4.171	TCP	70	49680 22 → 39744 [SYN, ACK] Seq=835172681 Ack=173	


```

> Frame 2: 70 bytes on wire (560 bits), 70 bytes captured (560 bits)
> Ethernet II, Src: Cisco_1f:72:4e (00:5d:73:1f:72:4e), Dst: Cisco_f8:19:ff (00:22:bd:f8:19:ff)
> 802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 102
> Internet Protocol Version 4, Src: 10.77.19.11, Dst: 10.11.4.171
v Transmission Control Protocol, Src Port: 22, Dst Port: 39744, Seq: 835172681, Ack: 1737026094, Len: 0
  Source Port: 22
  Destination Port: 39744
  [Stream index: 0]
  [TCP Segment Len: 0]
  Sequence number: 835172681
  [Next sequence number: 835172681]
  Acknowledgment number: 1737026094
  1000 .... = Header Length: 32 bytes (8)
> Flags: 0x012 (SYN, ACK)
  Window size value: 49680
  [Calculated window size: 49680]
  Checksum: 0xa91b [unverified]
  [Checksum Status: Unverified]
  Urgent pointer: 0
v Options: (12 bytes), Maximum segment size, No-Operation (NOP), Window scale, No-Operation (NOP), No-Operation (NOP), SA
  > TCP Option - Maximum segment size: 1380 bytes
  > TCP Option - No-Operation (NOP)
  > TCP Option - Window scale: 0 (multiply by 1)
  > TCP Option - No-Operation (NOP)

```

Op dit punt is het nodig om een opname op de server te nemen, te bevestigen dat het degene is die adverteert met vensterschaal = 0 en het opnieuw te configureren (controleer de serverdocumentatie hoe u dit doet).

Scenario 2. Snelle overdracht

Laten we nu eens kijken naar het goede scenario (snelle overdracht via hetzelfde netwerk):

Topologie:



De rentestroom:

SRC IP: 10.11.2.124

Dst IP: 172.25.18.134

Protocol: SFTP (FTP over SSH)

Opname op FTD LINA-motor inschakelen

<#root>

firepower#

capture CAPI int INSIDE buffer 33554432 match ip host 10.11.2.124 host 172.25.18.134

firepower#

capture CAPO int OUTSIDE buffer 33554432 match ip host 10.11.2.124 host 172.25.18.134

Ronde reistijd (RTT) Berekening: In dit geval is de RTT $\hat{=}$ 300 msec.

No.	Time	Source	Destination	Protocol	Length
1	0.000000	10.11.2.124	172.25.18.134	TCP	78
2	0.267006	172.25.18.134	10.11.2.124	TCP	78
3	0.000137	10.11.2.124	172.25.18.134	TCP	70
4	0.003784	10.11.2.124	172.25.18.134	SSHv2	91
5	0.266863	172.25.18.134	10.11.2.124	TCP	70
6	0.013580	172.25.18.134	10.11.2.124	SSHv2	91

Berekening TCP-venstergrootte: de server adverteert met een TCP-vensterschaalfactor van 7.

```
> Internet Protocol Version 4, Src: 172.25.18.134, Dst: 10.11.2.124
v Transmission Control Protocol, Src Port: 22, Dst Port: 57093, Seq: 661963571, Ack: 1770516295, Len: 0
  Source Port: 22
  Destination Port: 57093
  [Stream index: 0]
  [TCP Segment Len: 0]
  Sequence number: 661963571
  [Next sequence number: 661963571]
  Acknowledgment number: 1770516295
  1010 .... = Header Length: 40 bytes (10)
  > Flags: 0x012 (SYN, ACK)
  Window size value: 14480
  [Calculated window size: 14480]
  Checksum: 0x6497 [unverified]
  [Checksum Status: Unverified]
  Urgent pointer: 0
  v Options: (20 bytes), Maximum segment size, SACK permitted, Timestamps, No-Operation (NOP), Window scale
    > TCP Option - Maximum segment size: 1300 bytes
    > TCP Option - SACK permitted
    > TCP Option - Timestamps: TSval 390233290, TSecr 981659424
    > TCP Option - No-Operation (NOP)
    > TCP Option - Window scale: 7 (multiply by 128)
  > [SEQ/ACK analysis]
```

De grootte van het TCP-venster van de server is $\hat{=}$ 1600000 bytes:

No.	Time	Source	Destination	Protocol	Length	Window size value	Calculated window size	Info
23...	0.002579	172.25.18.134	10.11.2.124	TCP	70	12854	1645312	22 → 57093 [FIN, ACK]
23...	0.266847	172.25.18.134	10.11.2.124	TCP	70	12854	1645312	22 → 57093 [ACK] Seq=
23...	0.268089	172.25.18.134	10.11.2.124	SSHv2	198	12854	1645312	Server: Encrypted pack
23...	0.000076	172.25.18.134	10.11.2.124	SSHv2	118	12854	1645312	Server: Encrypted pack
23...	0.000351	172.25.18.134	10.11.2.124	SSHv2	118	12854	1645312	Server: Encrypted pack
23...	0.000092	172.25.18.134	10.11.2.124	TCP	70	12854	1645312	22 → 57093 [ACK] Seq=
23...	0.000015	172.25.18.134	10.11.2.124	TCP	70	12854	1645312	22 → 57093 [ACK] Seq=
23...	0.000091	172.25.18.134	10.11.2.124	TCP	70	12854	1645312	22 → 57093 [ACK] Seq=

Op basis van deze waarden geeft de formule voor bandbreedtevertraging het volgende:

$$1600000 * 8 / 0.3 = 43 \text{ Mbps maximale theoretische overdrachtssnelheid}$$

Situatie 6. Langzame TCP-overdracht (scenario 2)

Probleem Beschrijving: FTP bestandsoverdracht (download) via de firewall is langzaam.

Deze afbeelding toont de Topologie:



Beïnvloede stroom:

SRC IP: 192.168.2.220

Laatste IP: 192.168.1.220

Protocol: FTP

Capture Analysis

Schakel opnamen in op de FTD LINA engine.

```
<#root>
```

```
firepower#
```

```
capture CAPI type raw-data buffer 33554432 interface INSIDE match tcp host 192.168.2.220 host 192.168.1.220
```

```
firepower#
```

```
cap CAPO type raw-data buffer 33554432 interface OUTSIDE match tcp host 192.168.2.220 host 192.168.1.220
```

Selecteer een FTP-DATA-pakket en volg het FTP-gegevenskanaal op FTD INSIDE Capture (CAPI):

75	0.000412	192.168.2.220	192.168.1.220	TCP	66 54494 → 2388 [ACK] Seq=1884231612 Ack=2670018383
76	0.000518	192.168.1.220	192.168.2.220	FTP-DATA	(PASV) (RETR file15mb)
77	0.000061	192.168.1.220	192.168.2.220	FTP-DATA	(PASV) (RETR file15mb)
78	0.000046	192.168.1.220	192.168.2.220	FTP-DATA	not captured] FTP Data: 124
79	0.000015	192.168.1.220	192.168.2.220	FTP-DATA	(PASV) (RETR file15mb)
80	0.000107	192.168.2.220	192.168.1.220	TCP	q=1884231612 Ack=2670019631
81	0.000092	192.168.2.220	192.168.1.220	TCP	q=1884231612 Ack=2670020879
82	0.000091	192.168.2.220	192.168.1.220	TCP	4494 → 2388 [ACK] Seq=188423
83	0.000015	192.168.2.220	192.168.1.220	TCP	4494 → 2388 [ACK] Seq=188423
84	0.000321	192.168.1.220	192.168.2.220	FTP-DATA	(PASV) (RETR file15mb)
85	0.000061	192.168.1.220	192.168.2.220	FTP-DATA	(PASV) (RETR file15mb)
86	0.000153	192.168.2.220	192.168.1.220	TCP	4494 → 2388 [ACK] Seq=188423
87	0.000122	192.168.2.220	192.168.1.220	TCP	4494 → 2388 [ACK] Seq=188423
88	0.918415	192.168.1.220	192.168.2.220	TCP	88 → 54494 [ACK] Seq=2670020
89	0.000397	192.168.2.220	192.168.1.220	TCP	=2670027119
90	0.000869	192.168.1.220	192.168.2.220	FTP-DATA	e15mb)

De inhoud van de FTP-GEGEVENSstroom:

26	0.000000	192.168.2.220	192.168.1.220	TCP	74 54494 → 2388 [SYN] Seq=1884231611 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=3577288500 TS
28	1.026564	192.168.2.220	192.168.1.220	TCP	74 [TCP Retransmission] 54494 → 2388 [SYN] Seq=1884231611 Win=29200 Len=0 MSS=1460 SACK_PERM
29	1.981584	192.168.1.220	192.168.2.220	TCP	74 2388 → 54494 [SYN, ACK] Seq=2669989678 Ack=1884231612 Win=8192 Len=0 MSS=1260 WS=256 SACK
30	0.000488	192.168.2.220	192.168.1.220	TCP	66 54494 → 2388 [ACK] Seq=1884231612 Ack=2669989679 Win=29312 Len=0 TSval=3577291508 TSecr=4
34	0.001617	192.168.1.220	192.168.2.220	FTP-DATA	1314 FTP Data: 1248 bytes (PASV) (RETR file15mb)
35	0.000351	192.168.2.220	192.168.1.220	TCP	66 54494 → 2388 [ACK] Seq=1884231612 Ack=2669990927 Win=32128 Len=0 TSval=3577291510 TSecr=4
36	0.000458	192.168.1.220	192.168.2.220	FTP-DATA	1314 [TCP Previous segment not captured] FTP Data: 1248 bytes (PASV) (RETR file15mb)
37	0.000061	192.168.1.220	192.168.2.220	FTP-DATA	1314 FTP Data: 1248 bytes (PASV) (RETR file15mb)
38	0.000198	192.168.2.220	192.168.1.220	TCP	78 [TCP Window Update] 54494 → 2388 [ACK] Seq=1884231612 Ack=2669990927 Win=35072 Len=0 TSval
39	0.000077	192.168.2.220	192.168.1.220	TCP	78 [TCP Window Update] 54494 → 2388 [ACK] Seq=1884231612 Ack=2669990927 Win=37888 Len=0 TSval
40	0.309096	192.168.1.220	192.168.2.220	TCP	1314 [TCP Out-Of-Order] 2388 → 54494 [ACK] Seq=2669990927 Ack=1884231612 Win=66048 Len=1248 TS
41	0.000488	192.168.2.220	192.168.1.220	TCP	66 54494 → 2388 [ACK] Seq=1884231612 Ack=2669994671 Win=40832 Len=0 TSval=3577291820 TSecr=4
42	0.000489	192.168.1.220	192.168.2.220	FTP-DATA	1314 FTP Data: 1248 bytes (PASV) (RETR file15mb)
43	0.000045	192.168.1.220	192.168.2.220	FTP-DATA	1314 [TCP Previous segment not captured] FTP Data: 1248 bytes (PASV) (RETR file15mb)
44	0.000077	192.168.1.220	192.168.2.220	FTP-DATA	1314 FTP Data: 1248 bytes (PASV) (RETR file15mb)
45	0.000244	192.168.2.220	192.168.1.220	TCP	66 54494 → 2388 [ACK] Seq=1884231612 Ack=2669995919 Win=43776 Len=0 TSval=3577291821 TSecr=4
46	0.000030	192.168.2.220	192.168.1.220	TCP	78 [TCP Window Update] 54494 → 2388 [ACK] Seq=1884231612 Ack=2669995919 Win=48768 Len=0 TSval
47	0.000504	192.168.1.220	192.168.2.220	FTP-DATA	1314 FTP Data: 1248 bytes (PASV) (RETR file15mb)
48	0.000259	192.168.2.220	192.168.1.220	TCP	78 [TCP Window Update] 54494 → 2388 [ACK] Seq=1884231612 Ack=2669995919 Win=51584 Len=0 TSval
49	0.918126	192.168.1.220	192.168.2.220	TCP	1314 [TCP Out-Of-Order] 2388 → 54494 [ACK] Seq=2669995919 Ack=1884231612 Win=66048 Len=1248 TS
50	0.000900	192.168.2.220	192.168.1.220	TCP	66 54494 → 2388 [ACK] Seq=1884231612 Ack=2670000911 Win=54528 Len=0 TSval=3577292741 TSecr=4
51	0.000519	192.168.1.220	192.168.2.220	FTP-DATA	1314 FTP Data: 1248 bytes (PASV) (RETR file15mb)
52	0.000061	192.168.1.220	192.168.2.220	FTP-DATA	1314 FTP Data: 1248 bytes (PASV) (RETR file15mb)
53	0.000015	192.168.1.220	192.168.2.220	FTP-DATA	1314 [TCP Previous segment not captured] FTP Data: 1248 bytes (PASV) (RETR file15mb)
54	0.000015	192.168.1.220	192.168.2.220	FTP-DATA	1314 FTP Data: 1248 bytes (PASV) (RETR file15mb)
55	0.000199	192.168.2.220	192.168.1.220	TCP	66 54494 → 2388 [ACK] Seq=1884231612 Ack=2670002159 Win=57472 Len=0 TSval=3577292742 TSecr=4
56	0.000229	192.168.2.220	192.168.1.220	TCP	66 54494 → 2388 [ACK] Seq=1884231612 Ack=2670003407 Win=60288 Len=0 TSval=3577292742 TSecr=4
57	0.000183	192.168.1.220	192.168.2.220	FTP-DATA	1314 FTP Data: 1248 bytes (PASV) (RETR file15mb)
58	0.000106	192.168.2.220	192.168.1.220	TCP	78 [TCP Window Update] 54494 → 2388 [ACK] Seq=1884231612 Ack=2670003407 Win=65280 Len=0 TSval
59	0.000168	192.168.2.220	192.168.1.220	TCP	78 [TCP Window Update] 54494 → 2388 [ACK] Seq=1884231612 Ack=2670003407 Win=68224 Len=0 TSval
60	0.000000	192.168.1.220	192.168.2.220	FTP-DATA	1314 FTP Data: 1248 bytes (PASV) (RETR file15mb)

De CAPO-opnameinhoud:

31	0.000000	192.168.2.220	192.168.1.220	TCP	74	54494 → 2388 [SYN] Seq=2157030681 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=3577288500
33	1.026534	192.168.2.220	192.168.1.220	TCP	74	[TCP Retransmission] 54494 → 2388 [SYN] Seq=2157030681 Win=29200 Len=0 MSS=1460 SACK_PERM=1
34	1.981400	192.168.1.220	192.168.2.220	TCP	74	2388 → 54494 [SYN, ACK] Seq=2224316911 Ack=2157030682 Win=8192 Len=0 MSS=1260 HS=256 SACK_PERM=1
35	0.000610	192.168.2.220	192.168.1.220	TCP	66	54494 → 2388 [ACK] Seq=2157030682 Ack=2224316912 Win=29312 Len=0 TSval=3577291508 TSecr=4
38	0.001328	192.168.1.220	192.168.2.220	FTP-DATA	1314	FTP Data: 1248 bytes (PASV) (RETR file15mb)
40	0.000641	192.168.2.220	192.168.1.220	TCP	66	54494 → 2388 [ACK] Seq=2157030682 Ack=2224318160 Win=32128 Len=0 TSval=3577291510 TSecr=4
41	0.000381	192.168.1.220	192.168.2.220	FTP-DATA	1314	[TCP Previous segment not captured] FTP Data: 1248 bytes (PASV) (RETR file15mb)
42	0.000046	192.168.1.220	192.168.2.220	FTP-DATA	1314	FTP Data: 1248 bytes (PASV) (RETR file15mb)
43	0.000290	192.168.2.220	192.168.1.220	TCP	78	[TCP Window Update] 54494 → 2388 [ACK] Seq=2157030682 Ack=2224318160 Win=35072 Len=0 TSval=3577291510 TSecr=4
44	0.000076	192.168.2.220	192.168.1.220	TCP	78	[TCP Window Update] 54494 → 2388 [ACK] Seq=2157030682 Ack=2224318160 Win=37888 Len=0 TSval=3577291510 TSecr=4
45	0.309005	192.168.1.220	192.168.2.220	TCP	1314	[TCP Out-Of-Order] 2388 → 54494 [ACK] Seq=2224318160 Ack=2157030682 Win=66048 Len=1248 TSval=3577291510 TSecr=4
46	0.000580	192.168.2.220	192.168.1.220	TCP	66	54494 → 2388 [ACK] Seq=2157030682 Ack=2224321904 Win=40832 Len=0 TSval=3577291820 TSecr=4
47	0.000412	192.168.1.220	192.168.2.220	FTP-DATA	1314	FTP Data: 1248 bytes (PASV) (RETR file15mb)
48	0.000061	192.168.1.220	192.168.2.220	FTP-DATA	1314	[TCP Previous segment not captured] FTP Data: 1248 bytes (PASV) (RETR file15mb)
49	0.000076	192.168.1.220	192.168.2.220	FTP-DATA	1314	FTP Data: 1248 bytes (PASV) (RETR file15mb)
50	0.000290	192.168.2.220	192.168.1.220	TCP	66	54494 → 2388 [ACK] Seq=2157030682 Ack=2224323152 Win=43776 Len=0 TSval=3577291821 TSecr=4
51	0.000046	192.168.2.220	192.168.1.220	TCP	78	[TCP Window Update] 54494 → 2388 [ACK] Seq=2157030682 Ack=2224323152 Win=48768 Len=0 TSval=3577291821 TSecr=4
52	0.000412	192.168.1.220	192.168.2.220	FTP-DATA	1314	FTP Data: 1248 bytes (PASV) (RETR file15mb)
53	0.000351	192.168.2.220	192.168.1.220	TCP	78	[TCP Window Update] 54494 → 2388 [ACK] Seq=2157030682 Ack=2224323152 Win=51584 Len=0 TSval=3577291821 TSecr=4
54	0.918019	192.168.1.220	192.168.2.220	TCP	1314	[TCP Out-Of-Order] 2388 → 54494 [ACK] Seq=2224323152 Ack=2157030682 Win=66048 Len=1248 TSval=3577291821 TSecr=4
55	0.001007	192.168.2.220	192.168.1.220	TCP	66	54494 → 2388 [ACK] Seq=2157030682 Ack=2224328144 Win=54528 Len=0 TSval=3577292741 TSecr=4
56	0.000457	192.168.1.220	192.168.2.220	FTP-DATA	1314	FTP Data: 1248 bytes (PASV) (RETR file15mb)
57	0.000061	192.168.1.220	192.168.2.220	FTP-DATA	1314	FTP Data: 1248 bytes (PASV) (RETR file15mb)
58	0.000016	192.168.1.220	192.168.2.220	FTP-DATA	1314	[TCP Previous segment not captured] FTP Data: 1248 bytes (PASV) (RETR file15mb)
59	0.000000	192.168.1.220	192.168.2.220	FTP-DATA	1314	FTP Data: 1248 bytes (PASV) (RETR file15mb)
60	0.000274	192.168.2.220	192.168.1.220	TCP	66	54494 → 2388 [ACK] Seq=2157030682 Ack=2224329392 Win=57472 Len=0 TSval=3577292742 TSecr=4
61	0.000214	192.168.2.220	192.168.1.220	TCP	66	54494 → 2388 [ACK] Seq=2157030682 Ack=2224330640 Win=60288 Len=0 TSval=3577292742 TSecr=4
62	0.000122	192.168.1.220	192.168.2.220	FTP-DATA	1314	FTP Data: 1248 bytes (PASV) (RETR file15mb)
63	0.000168	192.168.2.220	192.168.1.220	TCP	78	[TCP Window Update] 54494 → 2388 [ACK] Seq=2157030682 Ack=2224330640 Win=65280 Len=0 TSval=3577292742 TSecr=4
64	0.000107	192.168.1.220	192.168.2.220	FTP-DATA	1314	FTP Data: 1248 bytes (PASV) (RETR file15mb)

Belangrijkste punten:

1. Er zijn TCP Out-of-Order (OOO)-pakketten.
2. Er is een TCP-hertransmissie.
3. Er is een indicatie van pakketverlies (gevalen pakketten).

Tip: sla de opnamen op terwijl u naar **Bestand > Opgegeven pakketten exporteren** navigeert. Sla vervolgens alleen het **weergegeven** pakketbereik op

Aanbevolen acties

De acties die in deze paragraaf worden opgesomd, hebben tot doel de kwestie verder af te zwakken.

Actie 1. Identificeer de locatie van het pakketverlies.

In gevallen als deze moet u simultane opnamen maken en de verdeel- en veroveringsmethode gebruiken om de netwerksegmenten te identificeren die pakketverlies veroorzaken. Vanuit het firewallstandpunt zijn er 3 belangrijke scenario's:

1. Het pakketverlies wordt veroorzaakt door de firewall zelf.
2. Het pakketverlies wordt veroorzaakt stroomafwaarts naar het firewallapparaat (richting van server aan cliënt).
3. Het pakketverlies wordt stroomopwaarts veroorzaakt naar het firewallapparaat (richting van de client naar de server).

Packet loss veroorzaakt door de firewall: om te weten te komen of het pakketverlies door de firewall is veroorzaakt, is het nodig om de toegangsopname te vergelijken met de uitgangsoopname. Er zijn heel veel manieren om 2 verschillende opnamen te vergelijken. Deze paragraaf laat een manier zien om deze taak uit te voeren.

Procedure om 2 te vergelijken vangt op om het pakketverlies te identificeren

Stap 1. Zorg ervoor dat de 2 opnamen pakketten bevatten vanuit hetzelfde tijdvenster. Dit betekent dat er geen pakketten in de ene opname mogen zijn die voor of na de andere opname zijn opgenomen. Er zijn een paar manieren om dit te doen:

- Controleer de waarden voor eerste en laatste IP-pakketidentificatie (ID).
- Controleer de eerste en laatste pakkettijdstempelwaarden.

In dit voorbeeld kunt u zien dat de eerste pakketten van elke opname dezelfde IP ID-waarden hebben:

No.	Time	Source	Destination	Protocol	Length	Identification	Info
1	2019-10-16 16:13:44.169394	192.168.2.220	192.168.1.220	TCP	74	0x0a34 (2612)	54494 → 2388 [SYN] Seq=1884231611 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TS...
2	2019-10-16 16:13:45.195958	192.168.2.220	192.168.1.220	TCP	74	0x0a35 (2613)	[TCP Retransmission] 54494 → 2388 [SYN] Seq=1884231611 Win=29200 Len=0 MS...
3	2019-10-16 16:13:47.177542	192.168.1.220	192.168.2.220	TCP	74	0x151f (5407)	2388 → 54494 [SYN, ACK] Seq=2669989678 Ack=1884231612 Win=8192 Len=0 MSS...
4	2019-10-16 16:13:47.178030	192.168.2.220	192.168.1.220	TCP	66	0x0a36 (2614)	
5	2019-10-16 16:13:47.179647	192.168.1.220	192.168.2.220	TCP	1314	0x1521 (5409)	
6	2019-10-16 16:13:47.179998	192.168.2.220	192.168.1.220	TCP	66	0x0a37 (2615)	
7	2019-10-16 16:13:47.180456	192.168.1.220	192.168.2.220	TCP	1314	0x1523 (5411)	
8	2019-10-16 16:13:47.180517	192.168.1.220	192.168.2.220	TCP	1314	0x1524 (5412)	
9	2019-10-16 16:13:47.180715	192.168.2.220	192.168.1.220	TCP	78	0x0a38 (2616)	
10	2019-10-16 16:13:47.180792	192.168.2.220	192.168.1.220	TCP	78	0x0a39 (2617)	
11	2019-10-16 16:13:47.489888	192.168.1.220	192.168.2.220	TCP	1314	0x1525 (5413)	
12	2019-10-16 16:13:47.490376	192.168.2.220	192.168.1.220	TCP	66	0x0a3a (2618)	
13	2019-10-16 16:13:47.490865	192.168.1.220	192.168.2.220	TCP	1314	0x1526 (5414)	
14	2019-10-16 16:13:47.490910	192.168.1.220	192.168.2.220	TCP	1314	0x1528 (5416)	
15	2019-10-16 16:13:47.490987	192.168.1.220	192.168.2.220	TCP	1314	0x1529 (5417)	
16	2019-10-16 16:13:47.491231	192.168.2.220	192.168.1.220	TCP	66	0x0a3b (2619)	
17	2019-10-16 16:13:47.491261	192.168.2.220	192.168.1.220	TCP	78	0x0a3c (2620)	
18	2019-10-16 16:13:47.491765	192.168.1.220	192.168.2.220	TCP	1314	0x152a (5418)	
19	2019-10-16 16:13:47.492024	192.168.2.220	192.168.1.220	TCP	78	0x0a3d (2621)	
20	2019-10-16 16:13:48.410150	192.168.1.220	192.168.2.220	TCP	1314	0x152e (5422)	
21	2019-10-16 16:13:48.411050	192.168.2.220	192.168.1.220	TCP	66	0x0a3e (2622)	
22	2019-10-16 16:13:48.411569	192.168.1.220	192.168.2.220	TCP	1314	0x152f (5423)	
23	2019-10-16 16:13:48.411630	192.168.1.220	192.168.2.220	TCP	1314	0x1530 (5424)	
24	2019-10-16 16:13:48.411645	192.168.1.220	192.168.2.220	TCP	1314	0x1532 (5426)	
25	2019-10-16 16:13:48.411660	192.168.1.220	192.168.2.220	TCP	1314	0x1533 (5427)	
26	2019-10-16 16:13:48.411859	192.168.2.220	192.168.1.220	TCP	66	0x0a3f (2623)	
27	2019-10-16 16:13:48.412088	192.168.2.220	192.168.1.220	TCP	66	0x0a40 (2624)	

> Frame 1: 74 bytes on wire (592 bits), 74 bytes captured (592 bits)

> Ethernet II, Src: Vmware_0b:e3:cb (00:0c:29:0b:e3:cb), Dst: Cisco_9d:89:97 (50:3d:e5:9d:89:97)

> Internet Protocol Version 4, Src: 192.168.2.220, Dst: 192.168.1.220

> Transmission Control Protocol, Src Port: 54494, Dst Port: 2388, Seq: 1884231611, Len: 0

Indien ze niet hetzelfde zijn, dan:

1. Vergelijk de tijdstempels vanaf het eerste pakket van elke opname.
2. Van de opname met de laatste Timestamp krijgt een filter van het verandert de Timestamp filter van ==naar >= (het eerste pakket) en <= (het laatste pakket), b.v.:

No.	Time	Source	Destination	Protocol	Length	Info
1	2019-10-16 16:13:43.244692	192.168.2.220	192.168.1.220	TCP	74	38400 → 21 [S
2	2019-10-16 16:13:43.245638	192.168.1.220	192.168.2.220	TCP	74	21 → 38400 [S
3	2019-10-16 16:13:43.245867	192.168.2.220	192.168.1.220	TCP	66	38400 → 21 [A

▼ Frame 2: 74 bytes on wire (592 bits), 74 bytes captured (592 bits)

Encapsulation type: Ethernet (1)

Arrival Time: Oct 16, 2019 16:13:43.245638000 seconds

[Time shift for this packet: 0.000000000 seconds]

Epoch Time: 1571235223.245638000 seconds

[Time delta from previous captured frame: 0.000000000 seconds]

[Time delta from previous displayed frame: 0.000000000 seconds]

[Time since reference or first frame: 0.000000000 seconds]

Frame Number: 2

Frame Length: 74 bytes (592 bits)

Capture Length: 74 bytes (592 bits)

(frame.time >= "okt 16, 2019 16:13:43.244692000") &(frame.time <= "okt 16, 2019 16:20:21.785130000")

3. Exporteer de gespecificeerde pakketten naar een nieuwe opname, selecteer **Bestand > Opgegeven pakketten exporteren** en sla de **weergegeven** pakketten op. Op dit punt moeten beide opnamen pakketten bevatten die hetzelfde tijdvenster beslaan. U kunt de vergelijking van de 2 opnamen nu starten.

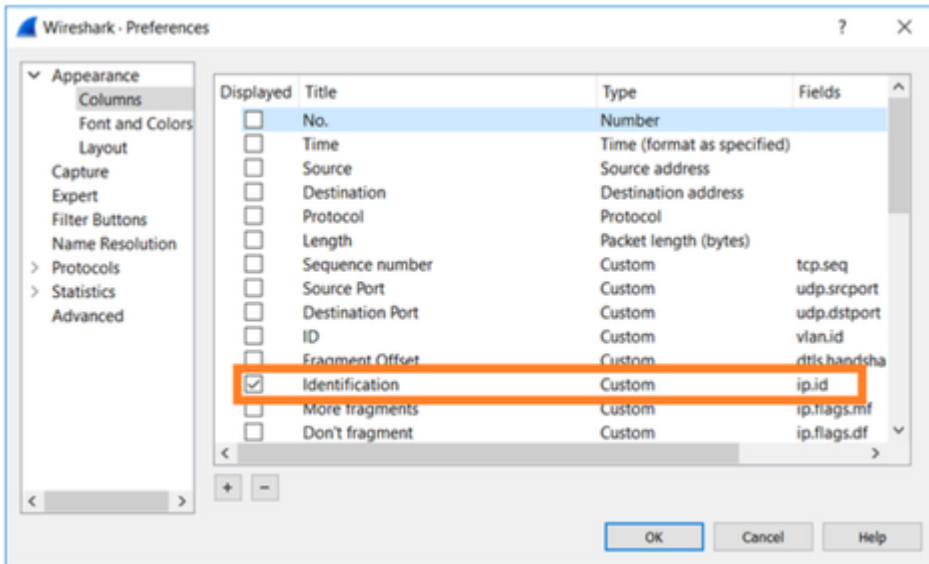
Stap 2. Specificeer welk pakketveld wordt gebruikt voor de vergelijking tussen de 2 opnamen. Voorbeeld van velden die kunnen worden gebruikt:

- IP-identificatie
- RTP-volnummer
- ICMP-volnummer

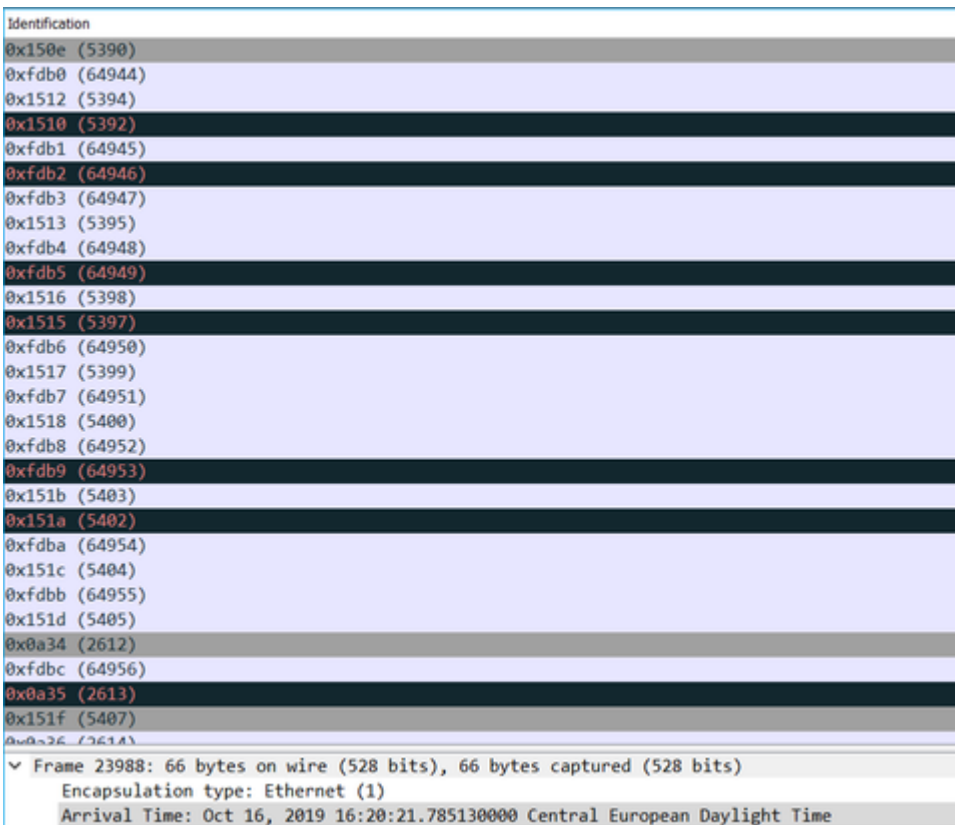
Maak een tekstversie van elke opname die het veld bevat voor elk pakket dat u in stap 1 hebt opgegeven. Om dit te doen, laat alleen de kolom van belang, bijvoorbeeld, als u pakketten wilt vergelijken die op IP Identificatie zijn gebaseerd, dan de opname wijzigen zoals in de afbeelding.

Apply a display filter ... <Ctrl-/>

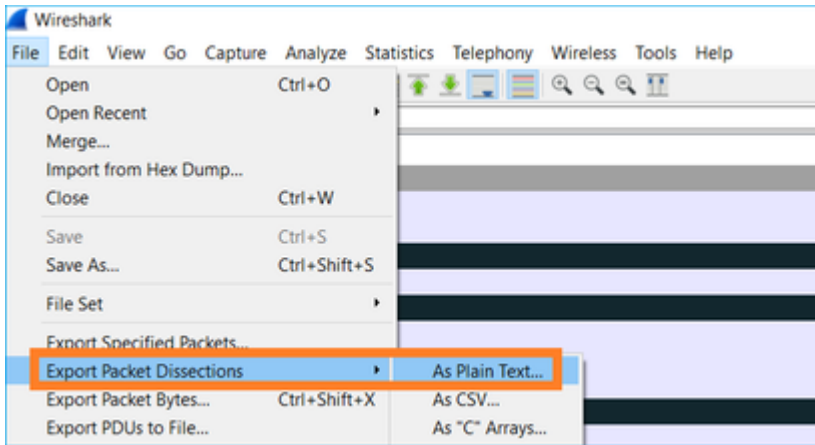
No.	Time	Source	Destination	Protocol	Length	Info
2	2019-10-16 16:13:43.245638	192.168.1.220	192.168.2.220	TCP	74	21 → 38400
3	2019-10-16 16:13:43.245867	192.168.2.220	192.168.1.220	TCP	66	38400 → 21
4	2019-10-16 16:13:43.558259	192.168.1.220	192.168.2.220	FTP	229	Response
5	2019-10-16 16:13:43.558274	192.168.1.220	192.168.2.220	TCP	126	[TCP Out



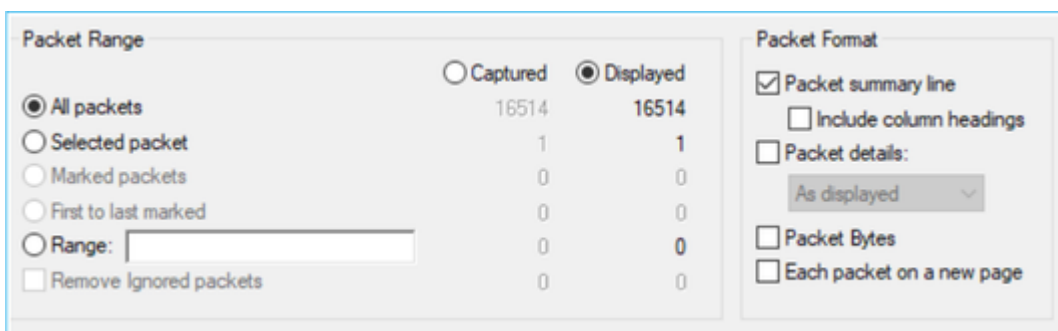
Het resultaat:



Stap 3. Maak een tekstversie van de opname (**Bestand > Verdelingen voor exportpakketten > Als onbewerkte tekst...**), zoals in de afbeelding:



Schakel de opties **Kolomkoppen** en **pakketdetails** opnemen uit om alleen de waarden van het weergegeven veld te exporteren, zoals in de afbeelding:



Stap 4. Sorteert de pakketten in de bestanden. U kunt de **opdracht** Linux **Sort** gebruiken om dit te doen:

```
<#root>
#
sort CAPI_IDs > file1.sorted
#
sort CAPO_IDs > file2.sorted
```

Stap 5. Gebruik een tekstvergelijkingstool (bijvoorbeeld WinMerge) of de opdracht Linux **diff** om de verschillen tussen de twee opnamen te vinden.

0x0a3d (2621)	0x0a3d (2621)
0x0a3e (2622)	0x0a3e (2622)
0x0a3f (2623)	0x0a3f (2623)
0x0a40 (2624)	0x0a40 (2624)
0x0a41 (2625)	0x0a41 (2625)
0x0a42 (2626)	0x0a42 (2626)
0x0a43 (2627)	0x0a43 (2627)
0x0a44 (2628)	0x0a44 (2628)
0x0a45 (2629)	0x0a45 (2629)
0x0a46 (2630)	0x0a46 (2630)
0x0a47 (2631)	0x0a47 (2631)
0x0a48 (2632)	0x0a48 (2632)
0x0a49 (2633)	0x0a49 (2633)
0x0a4a (2634)	0x0a4a (2634)
0x0a4b (2635)	0x0a4b (2635)
0x0a4c (2636)	0x0a4c (2636)
0x0a4d (2637)	0x0a4d (2637)
0x0a4e (2638)	0x0a4e (2638)
0x0a4f (2639)	0x0a4f (2639)

WinMerge

The selected files are identical.

Don't display this message again.

Ok

Ln: 27 Col: 14/14 Ch: 14/14 1252 Win Ln: 23955 Col: 1/1 Ch: 1/1

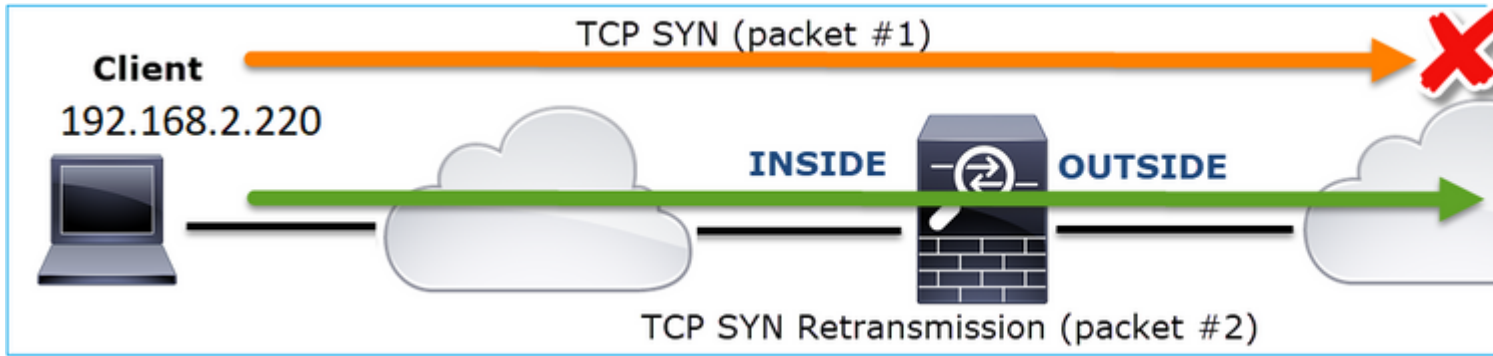
In dit geval zijn CAPI en CAPO Capo Capo voor het FTP Data Traffic identiek. Dit bewijst dat het pakketverlies niet door de firewall werd veroorzaakt.

Identificeer stroomopwaarts/stroomafwaarts pakketverlies.

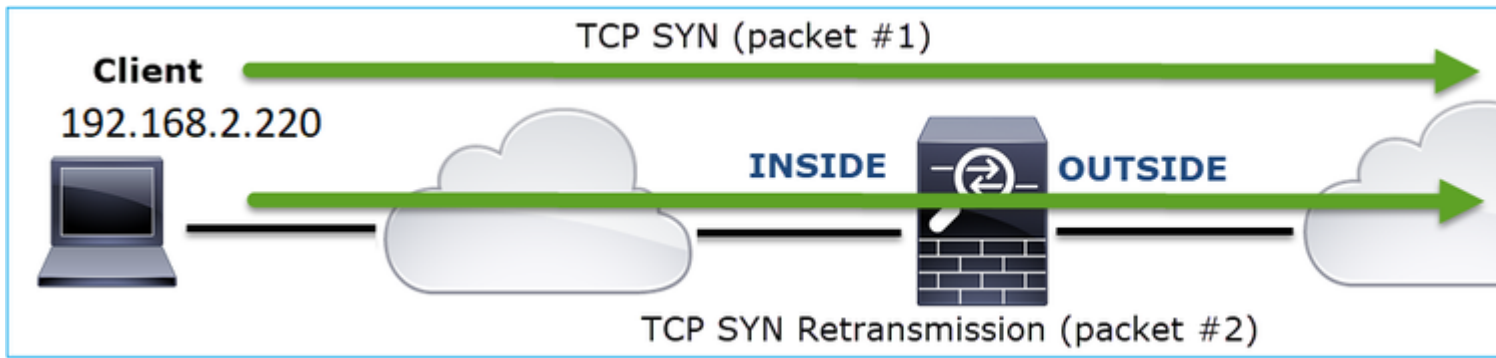
No.	Time	Source	Destination	Protocol	Length	Info
1	2019-10-16 16:13:44.169516	192.168.2.220	192.168.1.220	TCP	74	54494 → 2388 [SYN] Seq=2157030681 Win=29200 Len=0
2	2019-10-16 16:13:45.196050	192.168.2.220	192.168.1.220	TCP	74	[TCP Retransmission] 54494 → 2388 [SYN] Seq=2157030681
3	2019-10-16 16:13:47.177450	192.168.1.220	192.168.2.220	TCP	74	2388 → 54494 [SYN, ACK] Seq=2224316911 Ack=2157030682
4	2019-10-16 16:13:47.178060	192.168.2.220	192.168.1.220	TCP	66	54494 → 2388 [ACK] Seq=2157030682 Ack=2224316912
5	2019-10-16 16:13:47.179388	192.168.1.220	192.168.2.220	TCP	1314	2388 → 54494 [ACK] Seq=2224316912 Ack=2157030682
6	2019-10-16 16:13:47.180029	192.168.2.220	192.168.1.220	TCP	66	54494 → 2388 [ACK] Seq=2157030682 Ack=2224318160
7	2019-10-16 16:13:47.180410	192.168.1.220	192.168.2.220	TCP	1314	[TCP Previous segment not captured] 2388 → 54494 [ACK] Seq=2224318160
8	2019-10-16 16:13:47.180456	192.168.1.220	192.168.2.220	TCP	1314	2388 → 54494 [ACK] Seq=2224320656 Ack=2157030682
9	2019-10-16 16:13:47.180746	192.168.2.220	192.168.1.220	TCP	78	[TCP Window Update] 54494 → 2388 [ACK] Seq=2157030682
10	2019-10-16 16:13:47.180822	192.168.2.220	192.168.1.220	TCP	78	[TCP Window Update] 54494 → 2388 [ACK] Seq=2157030682
11	2019-10-16 16:13:47.489827	192.168.1.220	192.168.2.220	TCP	1314	[TCP Out-Of-Order] 2388 → 54494 [ACK] Seq=2224318160
12	2019-10-16 16:13:47.490407	192.168.2.220	192.168.1.220	TCP	66	54494 → 2388 [ACK] Seq=2157030682 Ack=2224321904
13	2019-10-16 16:13:47.490819	192.168.1.220	192.168.2.220	TCP	1314	2388 → 54494 [ACK] Seq=2224321904 Ack=2157030682
14	2019-10-16 16:13:47.490880	192.168.1.220	192.168.2.220	TCP	1314	[TCP Previous segment not captured] 2388 → 54494 [ACK] Seq=2224321904
15	2019-10-16 16:13:47.490956	192.168.1.220	192.168.2.220	TCP	1314	2388 → 54494 [ACK] Seq=2224325648 Ack=2157030682
16	2019-10-16 16:13:47.491246	192.168.2.220	192.168.1.220	TCP	66	54494 → 2388 [ACK] Seq=2157030682 Ack=2224323152

Belangrijkste punten:

1. Dit pakket is een TCP-hertransmissie. Het is met name een TCP/SYN-pakket dat van de client naar de server wordt verzonden voor FTP-gegevens in passieve modus. Aangezien de client het pakket opnieuw verstuurt en u het eerste SYN (#1) kunt zien, is het pakket stroomopwaarts verloren gegaan voor de firewall.



In dit geval is er de mogelijkheid dat het SYN-pakket naar de server is gekomen, maar het SYN/ACK-pakket is verloren op de terugweg:



2. Er is een pakket van de server en Wireshark identificeerde dat het vorige segment niet werd gezien/opgenomen. Aangezien het niet-opgenomen pakket van de server naar de client is verzonden en niet in de firewallopname is weergegeven, betekent dit dat het pakket tussen de server en de firewall is verloren.



Dit geeft aan dat er pakketverlies is tussen de FTP-server en de firewall.

Actie 2. Neem extra opnamen.

Neem extra opnamen samen met opnamen op de eindpunten. Probeer de verdeel- en verovermethode toe te passen om het problematische segment dat het pakketverlies veroorzaakt verder te isoleren.

No.	Time	Source	Destination	Protocol	Length	Info
155	2019-10-16 16:13:51.749845	192.168.1.220	192.168.2.220	FTP-DA...	1314	FTP Data: 1248 bytes (PASV)
156	2019-10-16 16:13:51.749860	192.168.1.220	192.168.2.220	FTP-DA...	1314	FTP Data: 1248 bytes (PASV)
157	2019-10-16 16:13:51.749872	192.168.1.220	192.168.2.220	FTP-DA...	1314	FTP Data: 1248 bytes (PASV)
158	2019-10-16 16:13:51.750722	192.168.2.220	192.168.1.220	TCP	66	54494 → 2388 [ACK] Seq=2157
159	2019-10-16 16:13:51.750744	192.168.1.220	192.168.2.220	FTP-DA...	1314	FTP Data: 1248 bytes (PASV)
160	2019-10-16 16:13:51.750768	192.168.2.220	192.168.1.220	TCP	66	54494 → 2388 [ACK] Seq=2157
161	2019-10-16 16:13:51.750782	192.168.1.220	192.168.2.220	FTP-DA...	1314	FTP Data: 1248 bytes (PASV)
162	2019-10-16 16:13:51.751001	192.168.2.220	192.168.1.220	TCP	78	[TCP Dup ACK 160#1] 54494 →
163	2019-10-16 16:13:51.751024	192.168.1.220	192.168.2.220	FTP-DA...	1314	FTP Data: 1248 bytes (PASV)
164	2019-10-16 16:13:51.751378	192.168.2.220	192.168.1.220	TCP	78	[TCP Dup ACK 160#2] 54494 →
165	2019-10-16 16:13:51.751402	192.168.1.220	192.168.2.220	FTP-DA...	1314	FTP Data: 1248 bytes (PASV)
166	2019-10-16 16:13:51.751622	192.168.2.220	192.168.1.220	TCP	78	[TCP Dup ACK 160#3] 54494 →
167	2019-10-16 16:13:51.751648	192.168.1.220	192.168.2.220	FTP-DA...	1314	[TCP Fast Retransmission]


```

> Frame 167: 1314 bytes on wire (10512 bits), 1314 bytes captured (10512 bits) on interface 0
> Ethernet II, Src: Vmware_30:2b:78 (00:0c:29:30:2b:78), Dst: Cisco_9d:89:9b (50:3d:e5:9d:89:9b)
> Internet Protocol Version 4, Src: 192.168.1.220, Dst: 192.168.2.220
> Transmission Control Protocol, Src Port: 2388, Dst Port: 494, Seq: 2224386800, Ack: 2157030682, Len: 1248
  FTP Data (1248 bytes data)
  [Setup frame: 33]
  [Setup method: PASV]
  [Command: RETR file15mb]
  Command frame: 40
  [Current working directory: /]
> Line-based text data (1 lines)

```

Belangrijkste punten:

1. De ontvanger (in dit geval de FTP-client) volgt de inkomende TCP-volnummers. Als het ontdekt dat een pakket werd gemist (een verwacht opeenvolgingsaantal werd overgeslagen) dan produceert het een ACK pakket met het ACK='verwachtte opeenvolgingsaantal dat werd overgeslagen'. In dit voorbeeld is de ACK=2224386800.
2. De Dup ACK activeert een TCP Fast Retransmission (hertransmissie binnen 20 msec nadat een dubbele ACK is ontvangen).

Wat betekenen Duplicate ACKs?

- Een paar dubbele ACKs maar geen daadwerkelijke wederuitzendingen wijzen erop dat er waarschijnlijker pakketten zijn die uit orde aankomen.
- Dubbele ACKs gevolgd door daadwerkelijke wederuitzendingen wijst erop dat er één of andere hoeveelheid pakketverlies is.

Actie 3. Bereken de verwerkingstijd van de firewall voor transitpakketten.

Pas de zelfde opname op 2 verschillende interfaces toe:

```

<#root>
firepower#
capture CAPI buffer 33554432 interface INSIDE match tcp host 192.168.2.220 host 192.168.1.220

firepower#
capture CAPI interface OUTSIDE

```

Exporteer de opnamecontrole op het tijdsverschil tussen de inkomende en de uitgaande pakketten

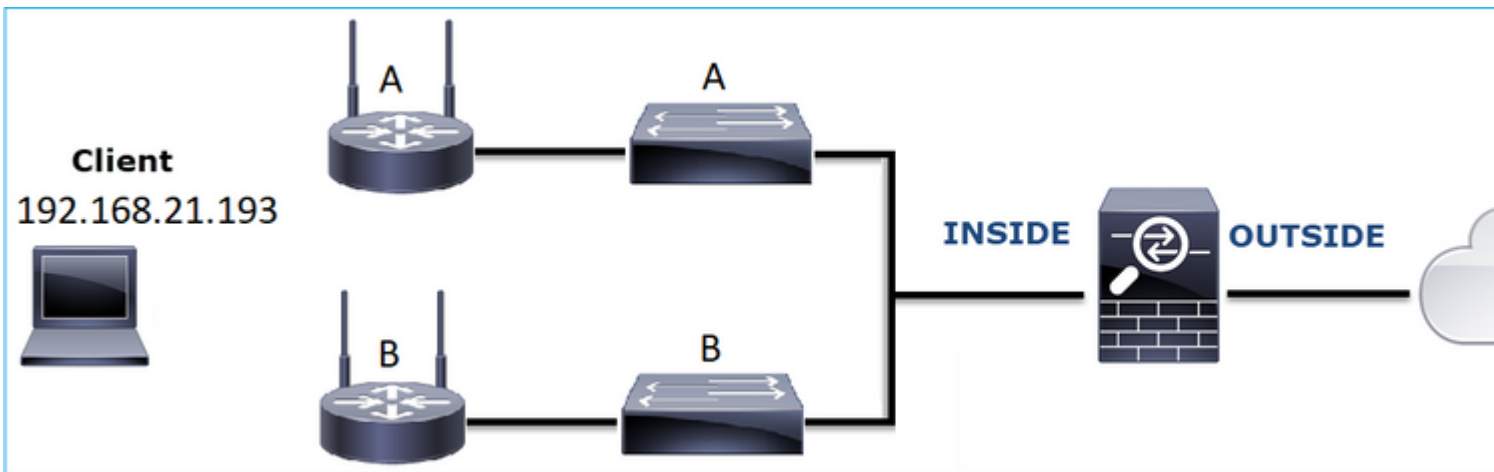
Situatie 7. Probleem met TCP-connectiviteit (pakketcorruptie)

Probleembeschrijving:

Draadloze client (192.168.21.193) probeert verbinding te maken met een doelserver (192.168.14.250 - HTTP) en er zijn 2 verschillende scenario's:

- Wanneer de client verbinding maakt met Access Point (AP) 'A' dan werkt de HTTP verbinding niet.
- Wanneer de client verbinding maakt met Access Point (AP) 'B', werkt de HTTP-verbinding.

Dit beeld toont de topologie:



Beïnvloede stroom:

SRC IP: 192.168.21.193

Laatste IP: 192.168.14.250

Protocol: TCP 80

Capture Analysis

Opnamen op FTD LINA-motor inschakelen:

```
<#root>
```

```
firepower#
```

```
capture CAPI int INSIDE match ip host 192.168.21.193 host 192.168.14.250
```

```
firepower#
```

capture CAPO int OUTSIDE match ip host 192.168.21.193 host 192.168.14.250

Opname - Functioneel scenario:

Als basislijn is het altijd erg handig om opnamen te maken van een bekend goed scenario.

Deze afbeelding toont de opname die is genomen op NGFW INSIDE interface

No.	Time	Source	Destination	Protocol	Length	Info
1	2013-08-08 17:03:25.554582	192.168.21.193	192.168.14.250	TCP	66	1055 → 80 [SYN] Seq=1341231 Win=65535 Len=0 MSS=
2	2013-08-08 17:03:25.555238	192.168.14.250	192.168.21.193	TCP	66	80 → 1055 [SYN, ACK] Seq=1015787006 Ack=1341232
3	2013-08-08 17:03:25.579910	192.168.21.193	192.168.14.250	TCP	58	1055 → 80 [ACK] Seq=1341232 Ack=1015787007 Win=6
4	2013-08-08 17:03:25.841081	192.168.21.193	192.168.14.250	HTTP	370	GET /ttest.html HTTP/1.1
5	2013-08-08 17:03:25.848466	192.168.14.250	192.168.21.193	TCP	1438	80 → 1055 [ACK] Seq=1015787007 Ack=1341544 Win=6
6	2013-08-08 17:03:25.848527	192.168.14.250	192.168.21.193	HTTP	698	HTTP/1.1 404 Not Found (text/html)
7	2013-08-08 17:03:25.858445	192.168.21.193	192.168.14.250	TCP	58	1055 → 80 [ACK] Seq=1341544 Ack=1015789027 Win=6
8	2013-08-08 17:03:34.391749	192.168.21.193	192.168.14.250	HTTP	369	GET /test.html HTTP/1.1
9	2013-08-08 17:03:34.395487	192.168.14.250	192.168.21.193	HTTP	586	HTTP/1.1 200 OK (text/html)
10	2013-08-08 17:03:34.606352	192.168.21.193	192.168.14.250	TCP	58	1055 → 80 [ACK] Seq=1341855 Ack=1015789555 Win=6
11	2013-08-08 17:03:40.739601	192.168.21.193	192.168.14.250	HTTP	483	GET /test.html HTTP/1.1
12	2013-08-08 17:03:40.741538	192.168.14.250	192.168.21.193	HTTP	271	HTTP/1.1 304 Not Modified

Deze afbeelding toont de opname die is genomen op NGFW BUITEN de interface.

No.	Time	Source	Destination	Protocol	Length	Info
1	2013-08-08 17:03:25.554872	192.168.21.193	192.168.14.250	TCP	66	1055 → 80 [SYN] Seq=1839800324 Win=65535 Len=0 MSS=
2	2013-08-08 17:03:25.555177	192.168.14.250	192.168.21.193	TCP	66	80 → 1055 [SYN, ACK] Seq=521188628 Ack=1839800325
3	2013-08-08 17:03:25.579926	192.168.21.193	192.168.14.250	TCP	58	1055 → 80 [ACK] Seq=1839800325 Ack=521188629 Win=6
4	2013-08-08 17:03:25.841112	192.168.21.193	192.168.14.250	HTTP	370	GET /ttest.html HTTP/1.1
5	2013-08-08 17:03:25.848451	192.168.14.250	192.168.21.193	TCP	1438	80 → 1055 [ACK] Seq=521188629 Ack=1839800637 Win=6
6	2013-08-08 17:03:25.848512	192.168.14.250	192.168.21.193	HTTP	698	HTTP/1.1 404 Not Found (text/html)
7	2013-08-08 17:03:25.858476	192.168.21.193	192.168.14.250	TCP	58	1055 → 80 [ACK] Seq=1839800637 Ack=521190649 Win=6
8	2013-08-08 17:03:34.391779	192.168.21.193	192.168.14.250	HTTP	369	GET /test.html HTTP/1.1
9	2013-08-08 17:03:34.395456	192.168.14.250	192.168.21.193	HTTP	586	HTTP/1.1 200 OK (text/html)
10	2013-08-08 17:03:34.606368	192.168.21.193	192.168.14.250	TCP	58	1055 → 80 [ACK] Seq=1839800948 Ack=521191177 Win=6
11	2013-08-08 17:03:40.739646	192.168.21.193	192.168.14.250	HTTP	483	GET /test.html HTTP/1.1
12	2013-08-08 17:03:40.741523	192.168.14.250	192.168.21.193	HTTP	271	HTTP/1.1 304 Not Modified

Belangrijkste punten:

1. De 2 opnamen zijn bijna identiek (overweeg de ISDN randomisering).
2. Er zijn geen aanwijzingen voor pakketverlies.
3. No Out-of-Order (OOO)-pakketten
4. Er zijn 3 HTTP GET aanvragen. De eerste krijgt een 404 "Not Found", de tweede krijgt een 200 "OK" en de derde een 304 "Not Modified" omleidingsbericht.

Captures - known-faulty scenario:

De inhoud van de toegangsoptname (CAPI).

No.	Time	Source	Destination	Protocol	Length	Info
1	2013-08-08 15:33:31.909193	192.168.21.193	192.168.14.250	TCP	66	3072 → 80 [SYN] Seq=4231766828 Win=65535 Len=0 MSS
2	2013-08-08 15:33:31.909849	192.168.14.250	192.168.21.193	TCP	66	80 → 3072 [SYN, ACK] Seq=867575959 Ack=4231766829
3	2013-08-08 15:33:31.913267	192.168.21.193	192.168.14.250	TCP	60	3072 → 80 [ACK] Seq=4231766829 Ack=867575960 Win=6
4	2013-08-08 15:33:31.913649	192.168.14.250	192.168.21.193	HTTP	222	HTTP/1.1 400 Bad Request (text/html)
5	2013-08-08 15:33:31.980326	192.168.21.193	192.168.14.250	TCP	369	[TCP Retransmission] 3072 → 80 [PSH, ACK] Seq=4231
6	2013-08-08 15:33:32.155723	192.168.14.250	192.168.21.193	TCP	58	[TCP ACKed unseen segment] 80 → 3072 [ACK] Seq=867
7	2013-08-08 15:33:34.871460	192.168.14.250	192.168.21.193	TCP	222	[TCP Retransmission] 80 → 3072 [FIN, PSH, ACK] Seq=
8	2013-08-08 15:33:34.894713	192.168.21.193	192.168.14.250	TCP	60	3072 → 80 [ACK] Seq=4231767140 Ack=867576125 Win=6
9	2013-08-08 15:33:34.933560	192.168.21.193	192.168.14.250	TCP	60	[TCP Retransmission] 3072 → 80 [FIN, ACK] Seq=4231
10	2013-08-08 15:33:34.933789	192.168.14.250	192.168.21.193	TCP	58	[TCP ACKed unseen segment] 80 → 3072 [ACK] Seq=867
11	2013-08-08 15:33:35.118234	192.168.21.193	192.168.14.250	TCP	66	3073 → 80 [SYN] Seq=2130836820 Win=65535 Len=0 MSS
12	2013-08-08 15:33:35.118737	192.168.14.250	192.168.21.193	TCP	66	80 → 3073 [SYN, ACK] Seq=2991287216 Ack=2130836821
13	2013-08-08 15:33:35.121575	192.168.21.193	192.168.14.250	TCP	60	3073 → 80 [ACK] Seq=2130836821 Ack=2991287217 Win=
14	2013-08-08 15:33:35.121621	192.168.21.193	192.168.14.250	TCP	371	[TCP Out-Of-Order] 3073 → 80 [PSH, ACK] Seq=213083
15	2013-08-08 15:33:35.121896	192.168.14.250	192.168.21.193	HTTP	222	HTTP/1.1 400 Bad Request (text/html)
16	2013-08-08 15:33:35.124657	192.168.21.193	192.168.14.250	TCP	60	3073 → 80 [ACK] Seq=2130837134 Ack=2991287382 Win=
17	2013-08-08 15:33:35.124840	192.168.14.250	192.168.21.193	TCP	58	[TCP ACKed unseen segment] 80 → 3073 [ACK] Seq=299
18	2013-08-08 15:33:35.126046	192.168.21.193	192.168.14.250	TCP	60	[TCP Spurious Retransmission] 3073 → 80 [FIN, ACK]
19	2013-08-08 15:33:35.126244	192.168.14.250	192.168.21.193	TCP	58	[TCP ACKed unseen segment] 80 → 3073 [ACK] Seq=299

Belangrijkste punten:

1. Er is een TCP 3-weg handdruk.
2. Er zijn TCP-heruitzendingen en aanwijzingen voor pakketverlies.
3. Er is een pakket (TCP ACK) dat door Wireshark wordt geïdentificeerd als **misvormd**.

Dit beeld toont de inhoud van de uitgaande opname (CAPO).

No.	Time	Source	Destination	Protocol	Length	Info
1	2013-08-08 15:33:31.909514	192.168.21.193	192.168.14.250	TCP	66	3072 → 80 [SYN] Seq=230342488 Win=65535 Len=0 MSS
2	2013-08-08 15:33:31.909804	192.168.14.250	192.168.21.193	TCP	66	80 → 3072 [SYN, ACK] Seq=268013986 Ack=230342489
3	2013-08-08 15:33:31.913298	192.168.21.193	192.168.14.250	TCP	60	3072 → 80 [ACK] Seq=230342489 Ack=268013987 Win=6
4	2013-08-08 15:33:31.913633	192.168.14.250	192.168.21.193	HTTP	222	HTTP/1.1 400 Bad Request (text/html)
5	2013-08-08 15:33:31.980357	192.168.21.193	192.168.14.250	TCP	369	[TCP Retransmission] 3072 → 80 [PSH, ACK] Seq=230
6	2013-08-08 15:33:32.155692	192.168.14.250	192.168.21.193	TCP	58	[TCP ACKed unseen segment] 80 → 3072 [ACK] Seq=26
7	2013-08-08 15:33:34.871430	192.168.14.250	192.168.21.193	TCP	222	[TCP Retransmission] 80 → 3072 [FIN, PSH, ACK] Se
8	2013-08-08 15:33:34.894759	192.168.21.193	192.168.14.250	TCP	60	3072 → 80 [ACK] Seq=230342800 Ack=268014152 Win=6
9	2013-08-08 15:33:34.933575	192.168.21.193	192.168.14.250	TCP	60	[TCP Retransmission] 3072 → 80 [FIN, ACK] Seq=230
10	2013-08-08 15:33:34.933774	192.168.14.250	192.168.21.193	TCP	58	[TCP ACKed unseen segment] 80 → 3072 [ACK] Seq=26
11	2013-08-08 15:33:35.118524	192.168.21.193	192.168.14.250	TCP	66	3073 → 80 [SYN] Seq=2731219422 Win=65535 Len=0 MS
12	2013-08-08 15:33:35.118707	192.168.14.250	192.168.21.193	TCP	66	80 → 3073 [SYN, ACK] Seq=2453407925 Ack=273121942
13	2013-08-08 15:33:35.121591	192.168.21.193	192.168.14.250	TCP	60	3073 → 80 [ACK] Seq=2731219423 Ack=2453407926 Win=
14	2013-08-08 15:33:35.121652	192.168.21.193	192.168.14.250	TCP	371	[TCP Out-Of-Order] 3073 → 80 [PSH, ACK] Seq=27312
15	2013-08-08 15:33:35.121865	192.168.14.250	192.168.21.193	HTTP	222	HTTP/1.1 400 Bad Request (text/html)
16	2013-08-08 15:33:35.124673	192.168.21.193	192.168.14.250	TCP	60	3073 → 80 [ACK] Seq=2731219736 Ack=2453408091 Win=
17	2013-08-08 15:33:35.124810	192.168.14.250	192.168.21.193	TCP	58	[TCP ACKed unseen segment] 80 → 3073 [ACK] Seq=24
18	2013-08-08 15:33:35.126061	192.168.21.193	192.168.14.250	TCP	60	[TCP Spurious Retransmission] 3073 → 80 [FIN, ACK]
19	2013-08-08 15:33:35.126229	192.168.14.250	192.168.21.193	TCP	58	[TCP ACKed unseen segment] 80 → 3073 [ACK] Seq=24

Belangrijkste punten:

De 2 opnamen zijn bijna identiek (overweeg de ISDN randomisering):

1. Er is een TCP 3-weg handdruk.
2. Er zijn TCP-heruitzendingen en aanwijzingen voor pakketverlies.
3. Er is een pakket (TCP ACK) dat door Wireshark wordt geïdentificeerd als **misvormd**.

Controleer het misvormde pakket:

No.	Time	Source	Destination	Protocol	Length	Info
1	2013-08-08 15:33:31.909193	192.168.21.193	192.168.14.250	TCP	66	3072 → 80 [SYN] Seq=4231766828 Win=65535 Len=
2	2013-08-08 15:33:31.909849	192.168.14.250	192.168.21.193	TCP	66	80 → 3072 [SYN, ACK] Seq=867575959 Ack=4231
3	2013-08-08 15:33:31.913267	192.168.21.193	192.168.14.250	TCP	60	3072 → 80 [ACK] Seq=4231766829 Ack=867575960


```

> Frame 3: 60 bytes on wire (480 bits), 60 bytes captured (480 bits)
> Ethernet II, Src: BelkinIn_63:90:f3 (ec:1a:59:63:90:f3), Dst: Cisco_61:cc:9b (58:8d:09:61:cc:9b)
> 802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 20
> Internet Protocol Version 4, Src: 192.168.21.193, Dst: 192.168.14.250
v Transmission Control Protocol, Src Port: 3072, Dst Port: 80, Seq: 4231766829, Ack: 867575960, Len: 2
  Source Port: 3072
  Destination Port: 80
  [Stream index: 0]
  [TCP Segment Len: 2]
  Sequence number: 4231766829
  [Next sequence number: 4231766831]
  Acknowledgment number: 867575960
  0101 .... = Header Length: 20 bytes (5)
  > Flags: 0x010 (ACK)
  Window size value: 65535
  [Calculated window size: 65535]
  [Window size scaling factor: -2 (no window scaling used)]
  Checksum: 0x01bf [unverified]
  [Checksum Status: Unverified]
  Urgent pointer: 0
  > [SEQ/ACK analysis]
  > [Timestamps]
  TCP payload (2 bytes)
v [Malformed Packet: Tunnel Socket]
  v [Expert Info (Error/Malformed): Malformed Packet (Exception occurred)]
    [Malformed Packet (Exception occurred)]
    [Severity level: Error]
    [Group: Malformed]

```

0000	58 8d 09 61 cc 9b ec 1a 59 63 90 f3 81 00 00 14	X..a....Yc.....
0010	08 00 45 00 00 2a 7f 1d 40 00 80 06 d5 a4 c0 a8	..E..*..@.....
0020	15 c1 c0 a8 0e fa 0c 00 00 50 fc 3b a7 0d 33 b6P;.-3.
0030	28 98 50 10 ff ff 01 bf 00 00 00 00	(.P.....

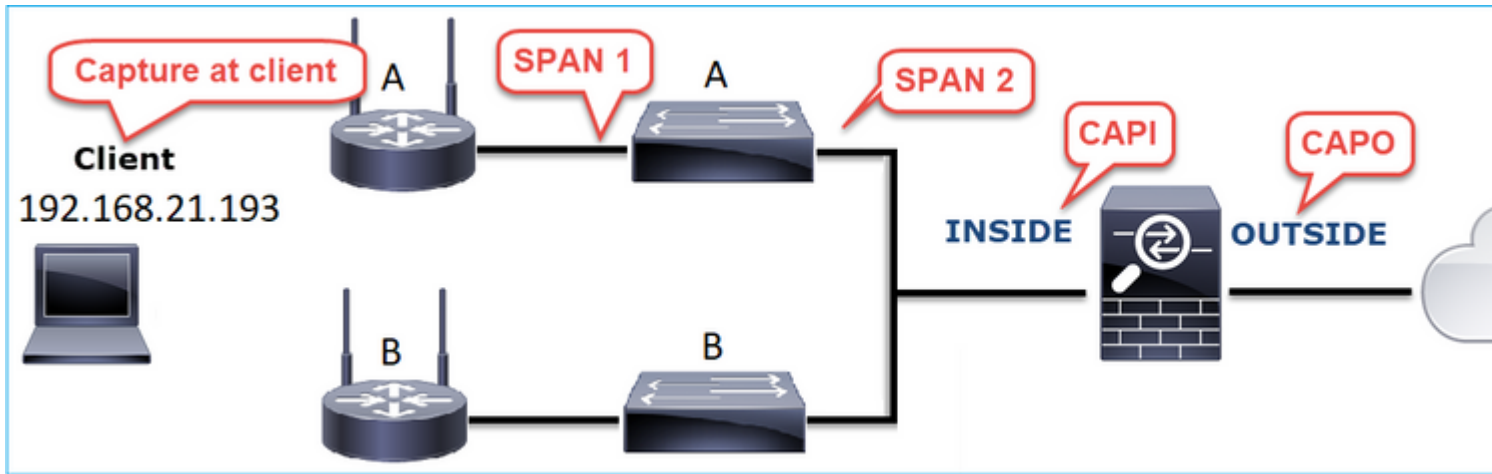
Belangrijkste punten:

1. Het pakket wordt geïdentificeerd als misvormd door Wireshark.
2. Het heeft een lengte van 2 bytes.
3. Er is een TCP payload van 2 Bytes.
4. De lading is 4 extra nullen (00 00).

Aanbevolen acties

De acties die in deze paragraaf worden opgesomd, hebben tot doel de kwestie verder af te zwakken.

Actie 1. Neem extra opnamen. Omvat opnamen op de eindpunten en probeer indien mogelijk de methode voor verdelen en veroveren toe te passen om de bron van de pakketcorruptie te isoleren, bijvoorbeeld:

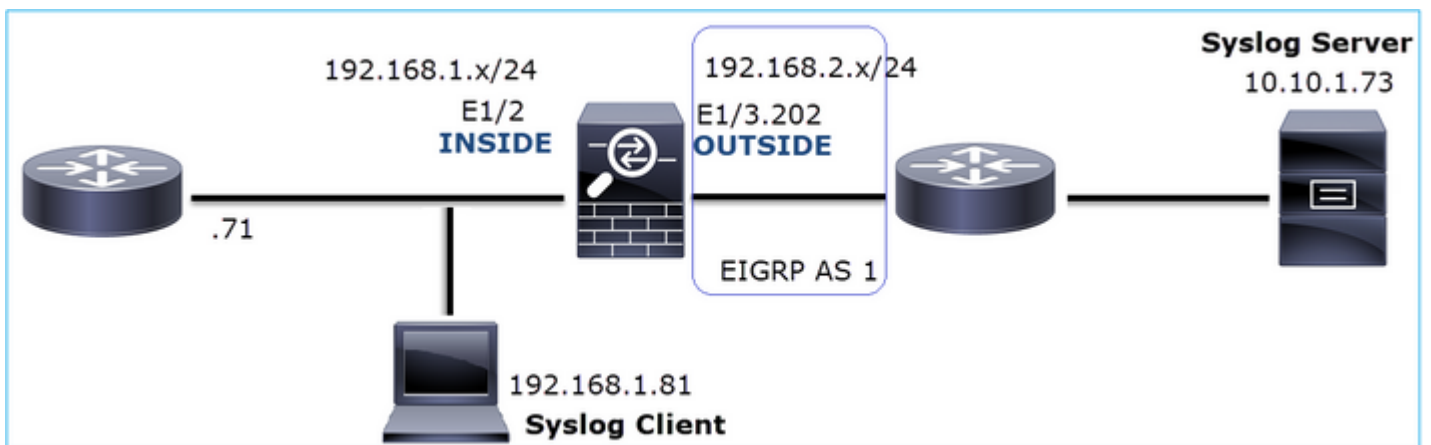


In dit geval werden de 2 extra bytes toegevoegd door de switch 'A'-interfacestuurprogramma en de oplossing was om de switch die de corruptie veroorzaakt te vervangen.

Situatie 8. UDP-connectiviteitsprobleem (ontbrekende pakketten)

Probleem Beschrijving: Syslog (UDP 514) berichten worden niet gezien op de bestemming Syslog server.

Dit beeld toont de topologie:



Beïnvloede stroom:

SRC IP: 192.168.1.81

Laatste IP: 10.10.1.73

Protocol: UDP 514

Capture Analysis

Opnamen op FTD LINA-motor inschakelen:

<#root>


```
firepower#
```

```
capture CAPI int INSIDE trace match udp host 192.168.1.81 host 10.10.1.73 eq 514
```

```
firepower#
```

```
capture CAPO int OUTSIDE match udp host 192.168.1.81 host 10.10.1.73 eq 514
```

FTD legt geen pakketten vast:

```
<#root>
```

```
firepower#
```

```
show capture
```

```
capture CAPI type raw-data trace interface INSIDE [Capturing - 0 bytes]
```

```
  match udp host 192.168.1.81 host 10.10.1.73 eq syslog
```

```
capture CAPO type raw-data interface OUTSIDE [Capturing - 0 bytes]
```

```
  match udp host 192.168.1.81 host 10.10.1.73 eq syslog
```

Aanbevolen acties

De acties die in deze paragraaf worden opgesomd, hebben tot doel de kwestie verder af te zwakken.

Actie 1. Controleer de FTD-verbindingstabel.

U kunt deze syntaxis gebruiken om een specifieke verbinding te controleren:

```
<#root>
```

```
firepower#
```

```
show conn address 192.168.1.81 port 514
```

```
10 in use, 3627189 most used
```

```
Inspect Snort:
```

```
  preserve-connection: 6 enabled, 0 in effect, 74 most enabled, 0 most in effect
```

```
UDP
```

```
INSIDE
```

```
  10.10.1.73:514
```

```
INSIDE
```

```
  192.168.1.81:514, idle 0:00:00, bytes
```

```
480379697
```

```
, flags -
```

```
o
```

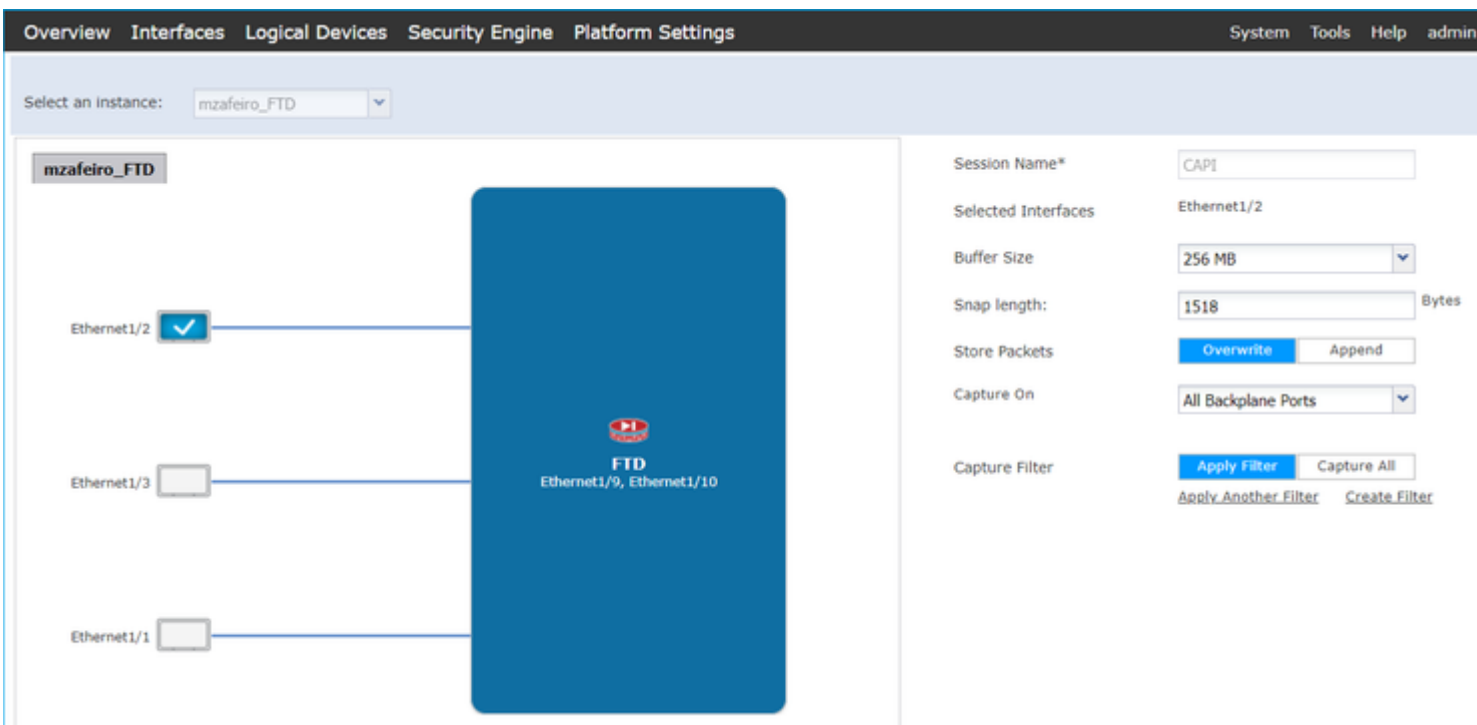
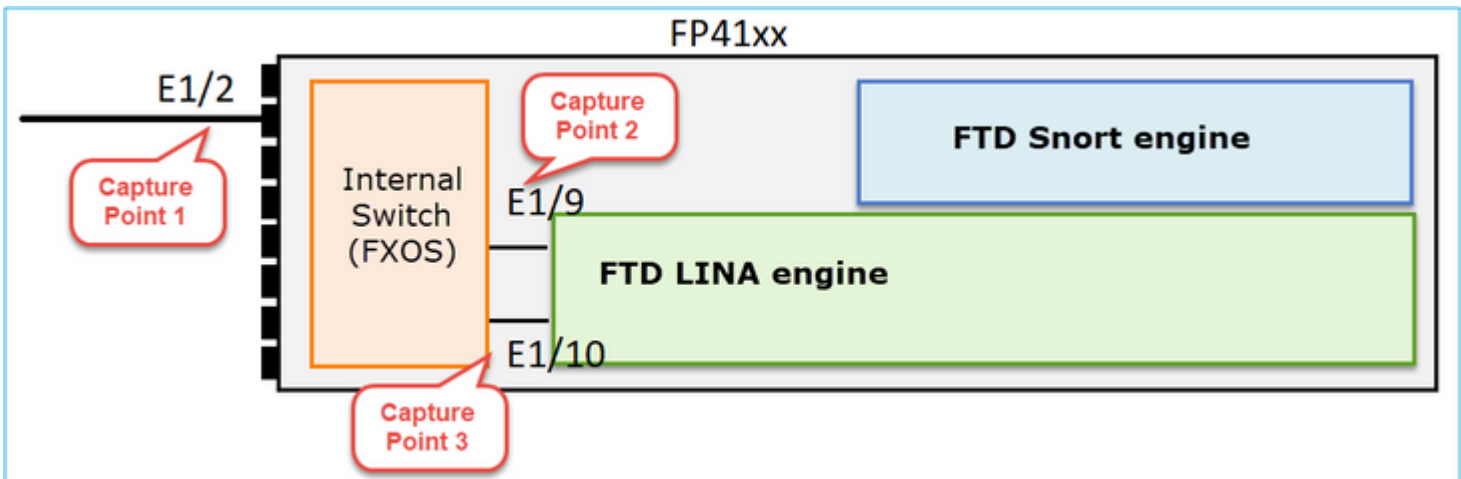
```
N1
```

Belangrijkste punten:

1. De in- en uitgangen zijn hetzelfde (bocht).
2. Het aantal bytes heeft een aanzienlijk grote waarde (ca. 5 GBytes).
3. De vlag "o" geeft de ontlasting (HW accelerated flow) aan. Dit is de reden waarom de FTD geen pakketten toont. Flow offload wordt alleen ondersteund op 41xx- en 93xx-platforms. In dit geval is het apparaat een 41xx.

Actie 2. Neem opnamen op chassinsniveau.

Maak verbinding met de Firepower chassis Manager en schakel de opname in op de ingangsiinterface (E1/2 in dit geval) en backplane interfaces (E1/9 en E1/10), zoals in de afbeelding:



Na enkele seconden:

Interface Name	Filter	File Size (in bytes)	File Name	Device Name
Ethernet1/10	None	276	CAPI-ethernet-1-10-0.pcap	mzafairo_FTD
Ethernet1/9	None	132276060	CAPI-ethernet-1-9-0.pcap	mzafairo_FTD
Ethernet1/2	None	136234072	CAPI-ethernet-1-2-0.pcap	mzafairo_FTD

Tip: in Wireshark sluit u de met VN gelabelde pakketten uit om pakketduplicatie op fysiek interfaceniveau te elimineren

Voor:

No.	Time	Source	Destination	Protocol	Length	Info
1	0.0000	Cisco_61:5a:9c	Spanning-tree-(f...	STP	64	RST. Root = 32768/0/00:11:bc:88:08:c9 Cost = 8 Port = 0x802d
2	0.0000	Cisco_61:5a:9c	Spanning-tree-(f...	STP	64	RST. Root = 32768/0/00:11:bc:88:08:c9 Cost = 8 Port = 0x802d
3	0.0532	Vmware_85:4f:ca	Broadcast	ARP	70	Who has 192.168.103.111? Tell 192.168.103.112
4	0.0000	Vmware_85:4f:ca	Broadcast	ARP	64	Who has 192.168.103.111? Tell 192.168.103.112
5	0.5216	Vmware_85:2f:00	Broadcast	ARP	70	Who has 10.10.10.1? Tell 10.10.10.10
6	0.0000	Vmware_85:2f:00	Broadcast	ARP	64	Who has 10.10.10.1? Tell 10.10.10.10
7	0.5770	Vmware_85:2f:00	Broadcast	ARP	70	Who has 10.10.10.1? Tell 10.10.10.10
8	0.0000	Vmware_85:2f:00	Broadcast	ARP	64	Who has 10.10.10.1? Tell 10.10.10.10
9	0.8479	Cisco_61:5a:9c	Spanning-tree-(f...	STP	64	RST. Root = 32768/0/00:11:bc:88:08:c9 Cost = 8 Port = 0x802d
10	0.0000	Cisco_61:5a:9c	Spanning-tree-(f...	STP	64	RST. Root = 32768/0/00:11:bc:88:08:c9 Cost = 8 Port = 0x802d
11	0.1520	Vmware_85:2f:00	Broadcast	ARP	70	Who has 10.10.10.1? Tell 10.10.10.10
12	0.0000	Vmware_85:2f:00	Broadcast	ARP	64	Who has 10.10.10.1? Tell 10.10.10.10
13	0.8606	Vmware_85:4f:ca	Broadcast	ARP	70	Who has 192.168.103.111? Tell 192.168.103.112
14	0.0000	Vmware_85:4f:ca	Broadcast	ARP	64	Who has 192.168.103.111? Tell 192.168.103.112
15	0.1655	192.168.0.101	173.38.200.100	DNS	91	Standard query 0x4a9f A 2.debian.pool.ntp.org
16	0.0000	192.168.0.101	173.38.200.100	DNS	85	Standard query 0x4a9f A 2.debian.pool.ntp.org
17	0.0000	192.168.0.101	173.38.200.100	DNS	91	Standard query 0x4afd AAAA 2.debian.pool.ntp.org
18	0.0000	192.168.0.101	173.38.200.100	DNS	85	Standard query 0x4afd AAAA 2.debian.pool.ntp.org
19	0.0003	192.168.0.101	173.38.200.100	DNS	91	Standard query 0x4a9f A 2.debian.pool.ntp.org
20	0.0000	192.168.0.101	173.38.200.100	DNS	85	Standard query 0x4a9f A 2.debian.pool.ntp.org

Na:

CAPI-ethernet-1-2-0.pcap

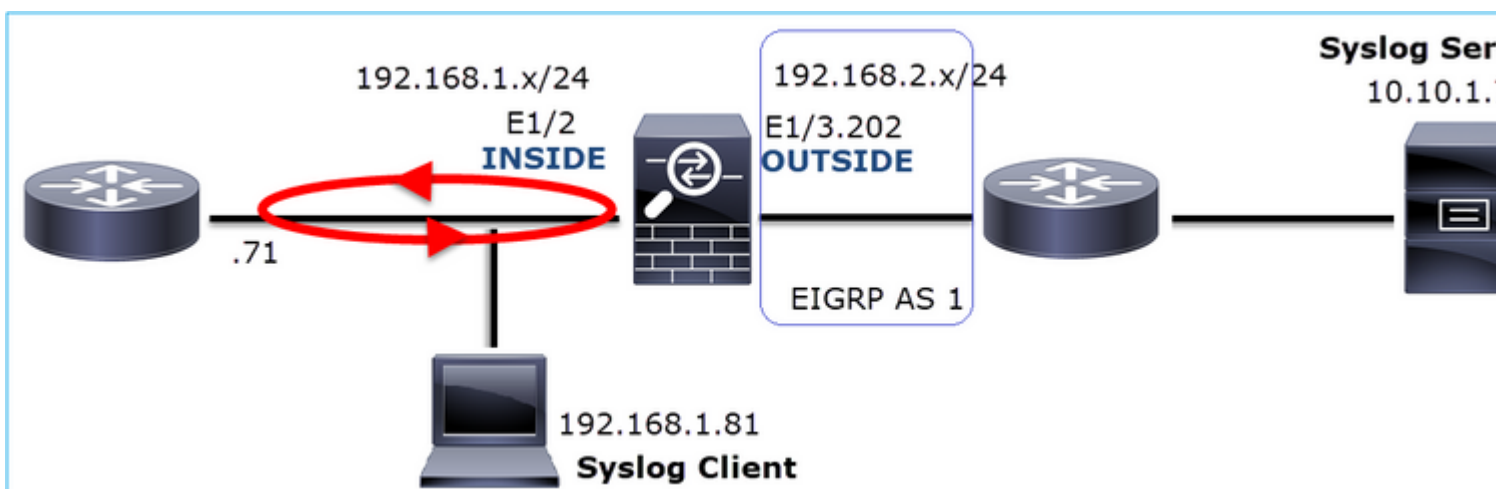
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

syslog && !vntag 1

No.	Time	Source	Destination	Protocol	Length	Time to live	Info
1334	0.000000000	192.168.1.81	10.10.1.73	Syslog	147	255	LOCAL4.DEBUG: Oct 15 2019 07:47:17: %ASA-7-609002
1336	0.00078873	192.168.1.81	10.10.1.73	Syslog	147	254	LOCAL4.DEBUG: Oct 15 2019 07:47:17: %ASA-7-609002
1338	0.00015099	192.168.1.81	10.10.1.73	Syslog	147	253	LOCAL4.DEBUG: Oct 15 2019 07:47:17: %ASA-7-609002
1340	0.000128919	192.168.1.81	10.10.1.73	Syslog	131	255	LOCAL4.DEBUG: Oct 15 2019 07:47:17: %ASA-7-609002
1342	0.000002839	192.168.1.81	10.10.1.73	Syslog	147	252	LOCAL4.DEBUG: Oct 15 2019 07:47:17: %ASA-7-609002
1344	0.000137974	192.168.1.81	10.10.1.73	Syslog	131	254	LOCAL4.DEBUG: Oct 15 2019 07:47:17: %ASA-7-609001
1346	0.000002758	192.168.1.81	10.10.1.73	Syslog	147	251	LOCAL4.DEBUG: Oct 15 2019 07:47:17: %ASA-7-609002
1348	0.000261845	192.168.1.81	10.10.1.73	Syslog	131	253	LOCAL4.DEBUG: Oct 15 2019 07:47:17: %ASA-7-609001
1350	0.000002736	192.168.1.81	10.10.1.73	Syslog	147	250	LOCAL4.DEBUG: Oct 15 2019 07:47:17: %ASA-7-609002
1352	0.000798149	192.168.1.81	10.10.1.73	Syslog	200	255	LOCAL4.INFO: Oct 15 2019 07:47:17: %ASA-6-302020:
1354	0.000498621	192.168.1.81	10.10.1.73	Syslog	131	252	LOCAL4.DEBUG: Oct 15 2019 07:47:17: %ASA-7-609001
1356	0.000002689	192.168.1.81	10.10.1.73	Syslog	147	249	LOCAL4.DEBUG: Oct 15 2019 07:47:17: %ASA-7-609002
1358	0.000697783	192.168.1.81	10.10.1.73	Syslog	195	255	LOCAL4.INFO: Oct 15 2019 07:47:17: %ASA-6-302021:
1360	0.000599702	192.168.1.81	10.10.1.73	Syslog	151	255	LOCAL4.DEBUG: Oct 15 2019 07:47:17: %ASA-7-609002
1362	0.000002728	192.168.1.81	10.10.1.73	Syslog	200	254	LOCAL4.INFO: Oct 15 2019 07:47:17: %ASA-6-302020:
1364	0.000499914	192.168.1.81	10.10.1.73	Syslog	131	251	LOCAL4.DEBUG: Oct 15 2019 07:47:17: %ASA-7-609001
1366	0.000697761	192.168.1.81	10.10.1.73	Syslog	147	248	LOCAL4.DEBUG: Oct 15 2019 07:47:17: %ASA-7-609002
1368	0.000169137	192.168.1.81	10.10.1.73	Syslog	195	254	LOCAL4.INFO: Oct 15 2019 07:47:17: %ASA-6-302021:
1370	0.000433196	192.168.1.81	10.10.1.73	Syslog	151	254	LOCAL4.DEBUG: Oct 15 2019 07:47:17: %ASA-7-609002
1372	0.000498718	192.168.1.81	10.10.1.73	Syslog	200	253	LOCAL4.INFO: Oct 15 2019 07:47:17: %ASA-6-302020:
1374	0.000002849	192.168.1.81	10.10.1.73	Syslog	131	250	LOCAL4.DEBUG: Oct 15 2019 07:47:17: %ASA-7-609001
1376	0.000596345	192.168.1.81	10.10.1.73	Syslog	147	247	LOCAL4.DEBUG: Oct 15 2019 07:47:17: %ASA-7-609002
1378	0.000600157	192.168.1.81	10.10.1.73	Syslog	195	253	LOCAL4.INFO: Oct 15 2019 07:47:17: %ASA-6-302021:
1380	0.000002772	192.168.1.81	10.10.1.73	Syslog	151	253	LOCAL4.DEBUG: Oct 15 2019 07:47:17: %ASA-7-609002
1382	0.000600947	192.168.1.81	10.10.1.73	Syslog	200	252	LOCAL4.INFO: Oct 15 2019 07:47:17: %ASA-6-302020:
1384	0.000498808	192.168.1.81	10.10.1.73	Syslog	131	249	LOCAL4.DEBUG: Oct 15 2019 07:47:17: %ASA-7-609001

Belangrijkste punten:

1. Er wordt een weergavefilter toegepast om pakketduplicaten te verwijderen en alleen syslogs weer te geven.
2. Het verschil tussen de pakketten is op het niveau van de microseconde. Dit duidt op een zeer hoge pakketsnelheid.
3. De waarde Time to Live (TTL) neemt voortdurend af. Dit geeft een pakketlus aan.



Actie 3. Gebruik de pakkettracer.

Aangezien de pakketten niet de firewall LINA engine doorkruisen, kunt u geen live-overtrekken (opnemen w/traceren) uitvoeren, maar u kunt een gemuleerd pakket overtrekken met packet-tracer:

<#root>

firepower#

packet-tracer input INSIDE udp 10.10.1.73 514 192.168.1.81 514

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 25350892, using existing flow

Phase: 4

Type: SNORT

Subtype:

Result: ALLOW

Config:

Additional Information:

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

Phase: 5

Type: ROUTE-LOOKUP

Subtype: Resolve Egress Interface

Result: ALLOW

Config:

Additional Information:

found next-hop 192.168.1.81 using egress ifc INSIDE

Phase: 6

Type: ADJACENCY-LOOKUP

Subtype: next-hop and adjacency

Result: ALLOW

Config:

Additional Information:

adjacency Active

next-hop mac address a023.9f92.2a4d hits 1 reference 1

Phase: 7

Type: CAPTURE

Subtype:

Result: ALLOW

Config:

Additional Information:

MAC Access list

Result:

```
input-interface: INSIDE
```

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

```
output-interface: INSIDE
```

```
output-status: up  
output-line-status: up  
Action: allow
```

Actie 4. Bevestig de FTD-routing.

Controleer de tabel met firewalls om te zien of er routeringsproblemen zijn:

```
<#root>
```

```
firepower#
```

```
show route 10.10.1.73
```

```
Routing entry for 10.10.1.0 255.255.255.0  
  Known via "eigrp 1", distance 90, metric 3072, type internal  
  Redistributing via eigrp 1  
  Last update from 192.168.2.72 on
```

```
OUTSIDE, 0:03:37 ago
```

```
Routing Descriptor Blocks:  
 * 192.168.2.72, from 192.168.2.72,
```

```
0:02:37 ago, via OUTSIDE
```

```
  Route metric is 3072, traffic share count is 1  
  Total delay is 20 microseconds, minimum bandwidth is 1000000 Kbit  
  Reliability 255/255, minimum MTU 1500 bytes  
  Loading 29/255, Hops 1
```

Belangrijkste punten:

1. De route wijst naar de juiste uitgang interface.
2. De route is een paar minuten geleden aangeleerd (0:02:37).

Actie 5. Bevestig de verbinding uptime.

Controleer de uptime van de verbinding om te zien wanneer deze verbinding tot stand is gebracht:

```
<#root>
```

```
firepower#
```

```
show conn address 192.168.1.81 port 514 detail
```

```
21 in use, 3627189 most used
```

```
Inspect Snort:
```

```
  preserve-connection: 19 enabled, 0 in effect, 74 most enabled, 0 most in effect
```

```
Flags: A - awaiting responder ACK to SYN, a - awaiting initiator ACK to SYN,
```

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

```
UDP INSIDE: 10.10.1.73/514 INSIDE: 192.168.1.81/514,  
  flags -oN1, idle 0s,
```

```
uptime 3m49s
```

```
, timeout 2m0s, bytes 4801148711
```

Belangrijkste punt:

1. De verbinding werd ~4 minuten geleden tot stand gebracht (dit is vóór de EIGRP-routeinstallatie in de routertabel)

Actie 6. Schakel de ingestelde verbinding uit.

In dit geval passen de pakketten een bestaande verbinding aan en worden ze naar een verkeerde uitloop-interface verstuurd; dit veroorzaakt een loop. Dit komt door de volgorde van de firewalls:

1. Opgezette verbindingsraadpleging (dit heeft voorrang op de wereldwijde raadpleging van routingstabel).
2. Network Address Translation (NAT) lookup - UN-NAT (bestemming NAT) fase krijgt voorrang boven PBR en routerlookup.
3. Op beleid gebaseerde routing (PBR)
4. Wereldwijde raadpleging van routingstabel

Aangezien de verbinding nooit keer uit (de Syslog-client stuurt voortdurend pakketten terwijl de UDP-conn-inactiviteitstimer 2 minuten is) is het nodig om de verbinding handmatig te wissen:

```
<#root>
```

```
firepower#
```

```
clear conn address 10.10.1.73 address 192.168.1.81 protocol udp port 514
```

```
1 connection(s) deleted.
```

Controleer of er een nieuwe verbinding tot stand is gebracht:

```

<#root>
firepower#
show conn address 192.168.1.81 port 514 detail | b 10.10.1.73.*192.168.1.81
UDP
OUTSIDE
: 10.10.1.73/514
INSIDE
: 192.168.1.81/514,
  flags -oN1, idle 1m15s, uptime 1m15s, timeout 2m0s, bytes 408

```

Actie 7. Configureer de drijvende-kommatimeout.

Dit is de juiste oplossing om het probleem aan te pakken en suboptimale routing te voorkomen, met name voor UDP-stromen. Navigeer naar **Apparaten > Platform-instellingen > Time-outs** en stel de waarde in:

SMTP Server	H.323	Default	0:0
SNMP	SIP	Default	0:3
SSL	SIP Media	Default	0:0
Syslog	SIP Disconnect:	Default	0:0
Timeouts	SIP Invite	Default	0:0
Time Synchronization	SIP Provisional Media	Default	0:0
UCAPL/CC Compliance	Floating Connection	Custom	0:0
	Xlate-PAT	Default	0:0

U vindt meer informatie over de drijvende-conn-time-out in de opdrachtreferentie:

<https://www.cisco.com/c/en/us/td/docs/security/asa/asa-command-reference/T-Z/cmdref4/t1.html#pgfid-1649892>

Situatie 9. Connectiviteitsprobleem met HTTPS (scenario 1)

Beschrijving van het probleem: Er kan geen HTTPS-communicatie tussen client 192.168.201.105 en server 192.168.202.101 worden vastgesteld

Dit beeld toont de topologie:



Beïnvloede stroom:

SRC IP: 192.168.201.11

Laatste IP: 192 168 202 111

Protocol: TCP 443 (HTTPS)

Capture Analysis

Opnamen op FTD LINA-motor inschakelen:

IP dat in de BUITENOPNAME wordt gebruikt, is anders vanwege de configuratie van de poortadresomzetting.

```
<#root>
```

```
firepower#
```

```
capture CAPI int INSIDE match ip host 192.168.201.111 host 192.168.202.111
```

```
firepower#
```

```
capture CAPO int OUTSIDE match ip host 192.168.202.111 host 192.168.202.111
```

Deze afbeelding toont de opname die is genomen op NGFW INSIDE interface:

No.	Time	Source	Destination	Protocol	Length	Identification	Info
38	2018-02-01 10:39:35.187887	192.168.201.111	192.168.202.111	TCP	78	0x2f31 (12081)	6666 → 443 [SYN] Seq=2034865631 Win=29200 Len=
39	2018-02-01 10:39:35.188909	192.168.202.111	192.168.201.111	TCP	78	0x0000 (0)	443 → 6666 [SYN, ACK] Seq=4086514531 Ack=20348
40	2018-02-01 10:39:35.189046	192.168.201.111	192.168.202.111	TCP	70	0x2f32 (12082)	6666 → 443 [ACK] Seq=2034865632 Ack=4086514532
41	2018-02-01 10:39:35.251695	192.168.201.111	192.168.202.111	TLSv1	326	0x2f33 (12083)	Client Hello
42	2018-02-01 10:39:35.252352	192.168.202.111	192.168.201.111	TCP	70	0xefb4 (61364)	443 → 6666 [ACK] Seq=4086514532 Ack=2034865888
43	2018-02-01 10:40:05.317320	192.168.202.111	192.168.201.111	TCP	70	0xd8c3 (55491)	443 → 6666 [RST] Seq=4086514532 Win=8192 Len=0

Belangrijkste punten:

1. Er is een TCP 3-weg handdruk.
2. SSL-onderhandeling start. De client stuurt een bericht van client-Hello.
3. Er wordt een TCP-ACK naar de client verzonden.

4. Er is een TCP RST verzonden naar de client.

Deze afbeelding toont de opname die is genomen op NGFW BUITEN de interface.

No.	Time	Source	Destination	Protocol	Length	Identification	Info
33	2018-02-01 10:39:35.188192	192.168.202.11	192.168.202.111	TCP	78	0x2f31 (12081)	15880 → 443 [SYN] Seq=2486930707 Win=29200 Len=0 MSS=1380
34	2018-02-01 10:39:35.188527	192.168.202.111	192.168.202.11	TCP	78	0x0000 (0)	443 → 15880 [SYN, ACK] Seq=3674405382 Ack=2486930708 Win=
35	2018-02-01 10:39:35.189214	192.168.202.11	192.168.202.111	TCP	70	0x2f32 (12082)	15880 → 443 [ACK] Seq=2486930708 Ack=3674405383 Win=29312
36	2018-02-01 10:39:35.252397	192.168.202.11	192.168.202.111	TLSv1	257	0xcd36 (52534)	Client Hello
37	2018-02-01 10:39:37.274430	192.168.202.11	192.168.202.111	TCP	257	0xb905 (47365)	[TCP Retransmission] 15880 → 443 [PSH, ACK] Seq=2486930708
38	2018-02-01 10:39:41.297332	192.168.202.11	192.168.202.111	TCP	257	0x88af (34991)	[TCP Retransmission] 15880 → 443 [PSH, ACK] Seq=2486930708
39	2018-02-01 10:39:49.309569	192.168.202.11	192.168.202.111	TCP	257	0xf68a (63114)	[TCP Retransmission] 15880 → 443 [PSH, ACK] Seq=2486930708
40	2018-02-01 10:40:05.317305	192.168.202.11	192.168.202.111	TCP	70	0xd621 (54817)	15880 → 443 [RST] Seq=2486930895 Win=8192 Len=0 TSval=192
41	2018-02-01 10:40:06.790700	192.168.202.111	192.168.202.11	TCP	78	0x0000 (0)	[TCP Retransmission] 443 → 15880 [SYN, ACK] Seq=367440538

Belangrijkste punten:

1. Er is een TCP 3-weg handdruk.
2. SSL-onderhandeling start. De client stuurt een bericht van client-Hello.
3. Er worden TCP-hertransmissies verzonden van de firewall naar de server.
4. Er is een TCP RST verzonden naar de server.

Aanbevolen acties

De acties die in deze paragraaf worden opgesomd, hebben tot doel de kwestie verder af te zwakken.

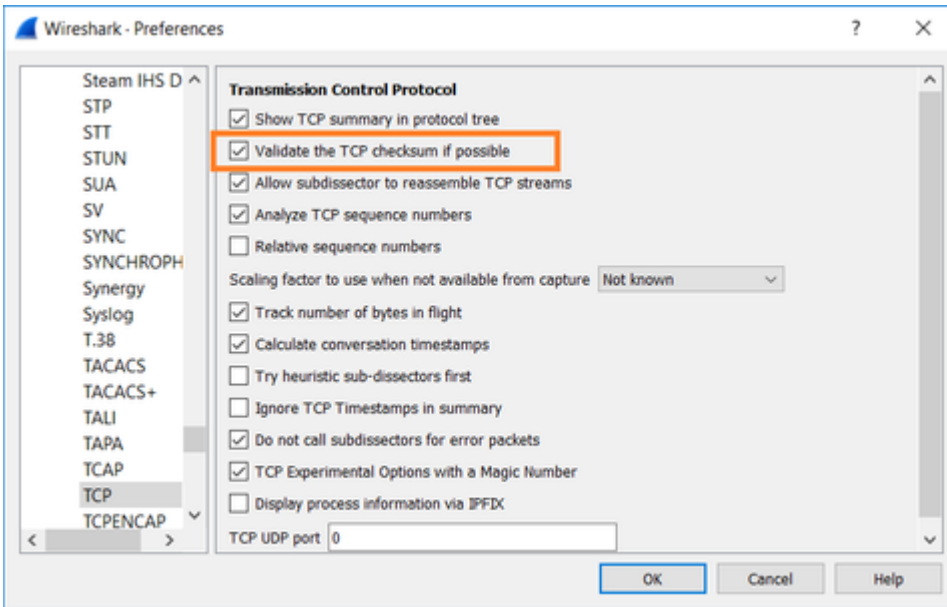
Actie 1. Neem extra opnamen.

Een opname op de server onthult dat de server de TLS-client Hellos met beschadigde TCP-checksum heeft ontvangen en deze stilzwijgend laat vallen (er is geen TCP RST of een ander antwoordpakket naar de client):

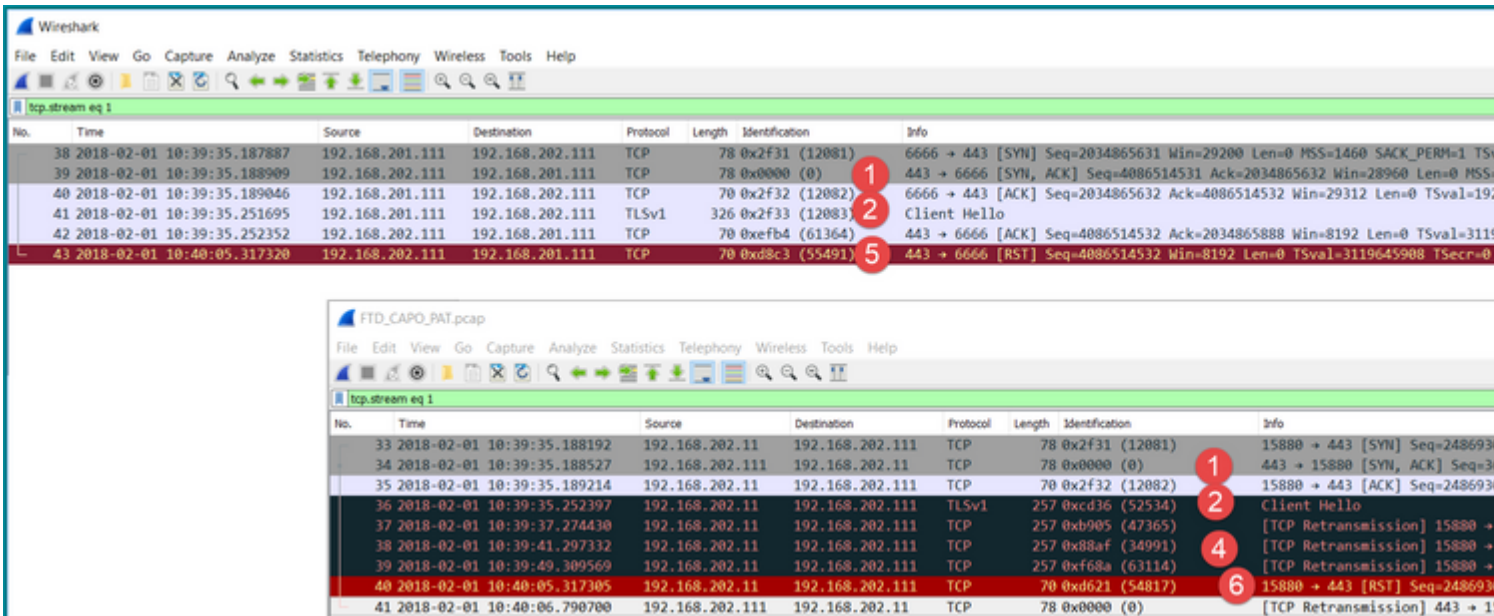
```
21:26:27.133677 IP (tos 0x0, ttl 64, id 52534, offset 0, flags [DF], proto TCP (6), length 239)
  192.168.202.11.15880 > 192.168.202.111.443: Flags [P.], cksum 0x0c65 (incorrect -> 0x3063), seq 1:188
S val 192658174 ecr 3119615816], length 187
21:26:29.155652 IP (tos 0x0, ttl 64, id 47365, offset 0, flags [DF], proto TCP (6), length 239)
  192.168.202.11.15880 > 192.168.202.111.443: Flags [P.], cksum 0x4db7 (incorrect -> 0x71b5), seq 1:188
S val 192660198 ecr 0], length 187
21:26:33.178142 IP (tos 0x0, ttl 64, id 34991, offset 0, flags [DF], proto TCP (6), length 239)
  192.168.202.11.15880 > 192.168.202.111.443: Flags [P.], cksum 0x3d1d (incorrect -> 0x61fb), seq 1:188
S val 192664224 ecr 0], length 187
21:26:41.189640 IP (tos 0x0, ttl 64, id 63114, offset 0, flags [DF], proto TCP (6), length 239)
  192.168.202.11.15880 > 192.168.202.111.443: Flags [P.], cksum 0x1e19 (incorrect -> 0x42a7), seq 1:188
S val 192672244 ecr 0], length 187
21:26:57.195947 IP (tos 0x0, ttl 64, id 54817, offset 0, flags [DF], proto TCP (6), length 52)
  192.168.202.11.15880 > 192.168.202.111.443: Flags [R], cksum 0x9ee1 (incorrect -> 0xc2e8), seq 248693
al 192688266 ecr 0], length 0
21:26:58.668973 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto TCP (6), length 60)
  192.168.202.111.443 > 192.168.202.11.15880: Flags [S.], cksum 0x15fb (incorrect -> 0xffd2), seq 36744
ptions [mss 1460,sackOK,TS val 3119647415 ecr 192658158,nop,wscale 7], length 0
^C
154 packets captured
154 packets received by filter
```

Als je alles samenvoegt:

In dit geval, om te begrijpen, is er een behoefte om op Wireshark de **Validate de controlesom van TCP** toe te laten **als mogelijke** optie. Navigeer om > **Voorkeuren** > **Protocollen** > **TCP** te bewerken, zoals in de afbeelding wordt getoond.



In dit geval is het handig om de opnamen naast elkaar te zetten om een volledig beeld te krijgen:



Belangrijkste punten:

1. Er is een TCP 3-weg handdruk. De IP-id's zijn hetzelfde. Dit betekent dat de stroom niet door de firewall is geproxyed.
2. Een TLS-client komt van de client met IP-ID 12083. Het pakket is geproxyed door de firewall (de firewall is in dit geval geconfigureerd met TLS-decryptie beleid) en de IP-id is gewijzigd in 52534. Bovendien wordt de TCP-checksum van het pakket beschadigd (door een softwarestoring die later is verholpen).
3. De firewall is in TCP Proxy modus en stuurt een ACK naar de client (die de server spoofs).

```

33 2018-02-01 10:39:35.188192 192.168.202.11 192.168.202.111 TCP 78 0x2f31 (12081) 15880 → 443 [SYN] Seq=2486930707 Win=29200 Len=0 MSS=1380 S
34 2018-02-01 10:39:35.188527 192.168.202.111 192.168.202.11 TCP 78 0x0000 (0) 443 → 15880 [SYN, ACK] Seq=3674405382 Ack=2486930708 Min=20
35 2018-02-01 10:39:35.189214 192.168.202.11 192.168.202.111 TCP 70 0x2f32 (12082) 15880 → 443 [ACK] Seq=2486930708 Ack=3674405383 Win=29312 L
36 2018-02-01 10:39:35.252397 192.168.202.11 192.168.202.111 TLSv1 257 0xcd36 (52534) Client Hello

```

```

> Internet Protocol Version 4, Src: 192.168.202.11, Dst: 192.168.202.111
  Transmission Control Protocol, Src Port: 15880, Dst Port: 443, Seq: 2486930708, Ack: 3674405383, Len: 187
    Source Port: 15880
    Destination Port: 443
    [Stream index: 1]
    [TCP Segment Len: 187]
    Sequence number: 2486930708
    [Next sequence number: 2486930895]
    Acknowledgment number: 3674405383
    1000 .... = Header Length: 32 bytes (8)
  > Flags: 0x018 (PSH, ACK)
    Window size value: 64
    [Calculated window size: 8192]
    [Window size scaling factor: 128]
  > Checksum: 0x0c65 incorrect, should be 0x3063(maybe caused by "TCP checksum offload"?)
    [Checksum Status: Bad]
    [Calculated Checksum: 0x3063]
    Urgent pointer: 0
  > Options: (12 bytes), No-Operation (NOP), No-Operation (NOP), Timestamps
  > [SEQ/ACK analysis]
  > [Timestamps]
  > TCP payload (187 bytes)
  > Secure Sockets Layer

```

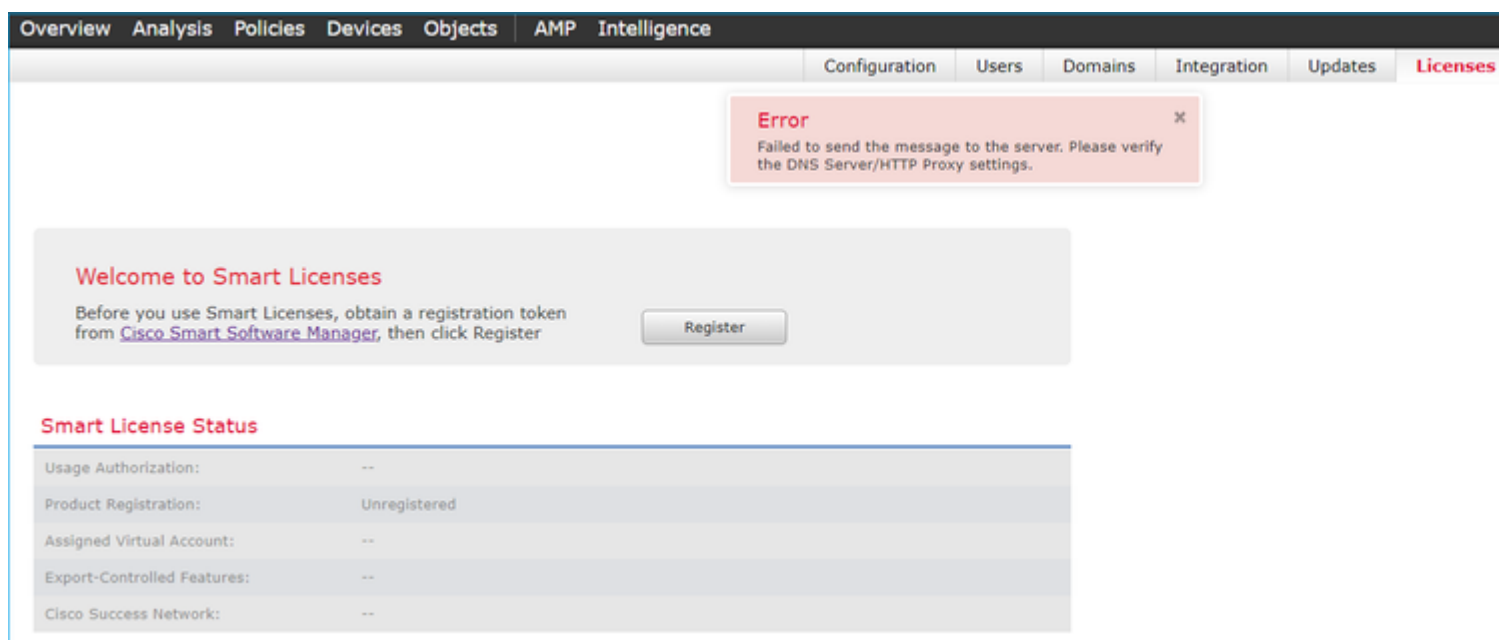
4. De firewall ontvangt geen TCP-ACK-pakket van de server en stuurt het bericht van TLS Client Hello opnieuw door. Dit is opnieuw te wijten aan TCP Proxy modus dat de firewall geactiveerd.
5. Na ~30 seconden geeft de firewall het op en stuurt een TCP RST naar de client.
6. De firewall stuurt een TCP/RST naar de server.

Ter referentie:

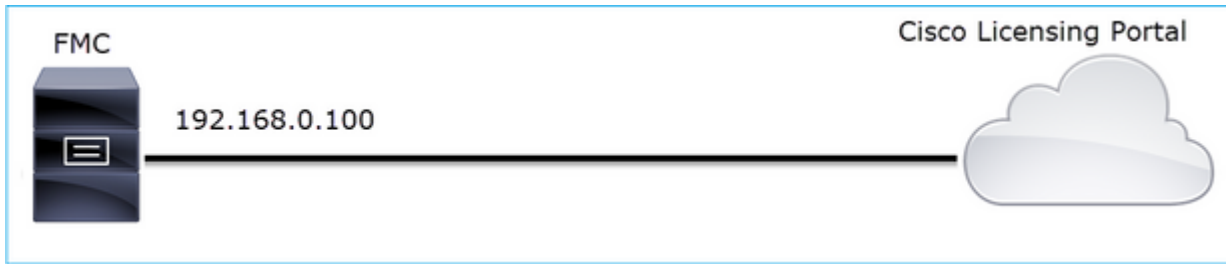
[Firepower TLS/SSL-handshake verwerking](#)

Situatie 10. Connectiviteitsprobleem met HTTPS (scenario 2)

Probleembeschrijving: registratie van Slimme FMC-licentie mislukt.



Dit beeld toont de topologie:



Beïnvloede stroom:

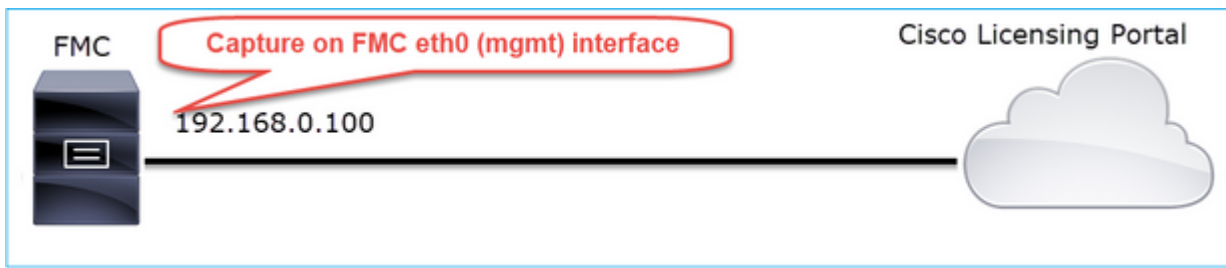
SRC IP: 192.168.0.10

Datum: tools.cisco.com

Protocol: TCP 443 (HTTPS)

Capture Analysis

Opname op de FMC-beheerinterface inschakelen:



Probeer je opnieuw te registreren. Zodra de foutmelding wordt weergegeven, drukt u op CTRL-C om de opname te stoppen:

```
<#root>
```

```
root@firepower:/Volume/home/admin#
```

```
tcpdump -i eth0 port 443 -s 0 -w CAP.pcap
```

```
HS_PACKET_BUFFER_SIZE is set to 4.
```

```
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
```

```
^C
```

```
264 packets captured
```

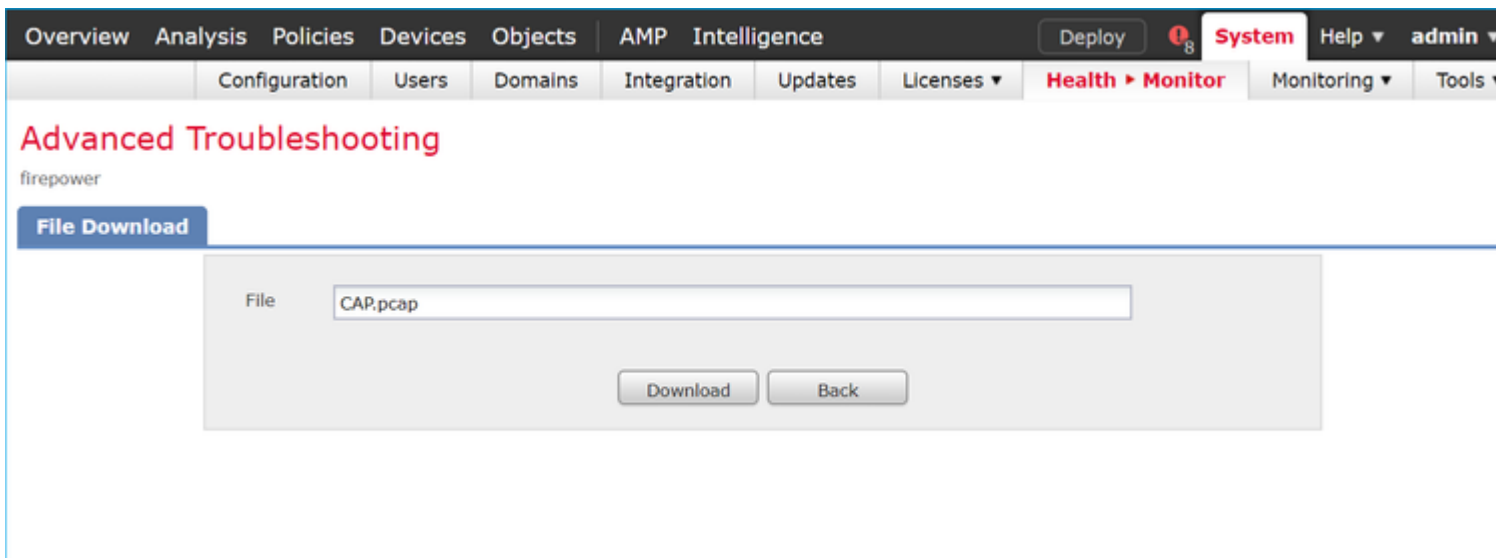
```
<- CTRL-C
```

```
264 packets received by filter
```

```
0 packets dropped by kernel
```

```
root@firepower:/Volume/home/admin#
```

Verzamel de opname van het VCC (**Systeem > Gezondheid > Monitor**, selecteer het apparaat en selecteer **Geavanceerde probleemoplossing**), zoals getoond in het beeld:



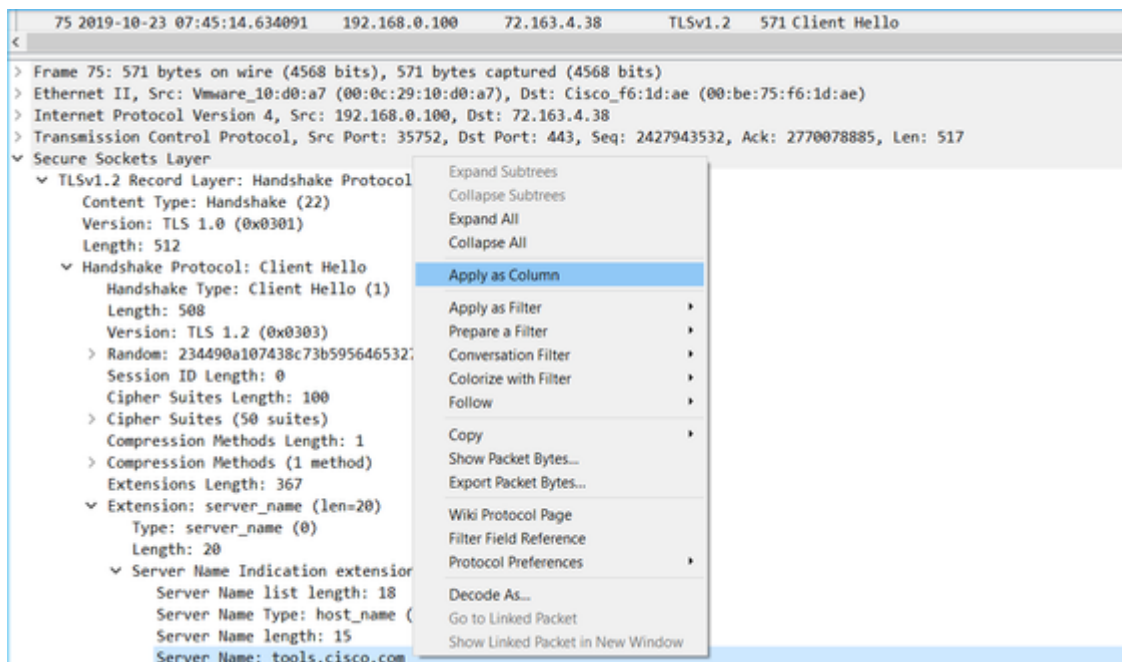
De afbeelding toont het VCC op Wireshark vastlegt:

No.	Time	Source	Destination	Protocol	Length	Info
1	2019-10-23 07:44:59.218797	192.168.0.100	10.229.20.96	TLSv1.2	107	Application Data
2	2019-10-23 07:44:59.220929	10.229.20.96	192.168.0.100	TLSv1.2	123	Application Data
3	2019-10-23 07:44:59.220960	192.168.0.100	10.229.20.96	TCP	54	443 → 64722 [ACK] Seq=13809
4	2019-10-23 07:45:02.215376	192.168.0.100	10.229.20.96	TLSv1.2	107	Application Data
5	2019-10-23 07:45:02.217321	10.229.20.96	192.168.0.100	TLSv1.2	123	Application Data
6	2019-10-23 07:45:02.217336	192.168.0.100	10.229.20.96	TCP	54	443 → 64722 [ACK] Seq=13809
7	2019-10-23 07:45:05.215460	192.168.0.100	10.229.20.96	TLSv1.2	107	Application Data
8	2019-10-23 07:45:05.217331	10.229.20.96	192.168.0.100	TLSv1.2	123	Application Data
9	2019-10-23 07:45:05.217345	192.168.0.100	10.229.20.96	TCP	54	443 → 64722 [ACK] Seq=13809
10	2019-10-23 07:45:06.216584	10.229.20.96	192.168.0.100	TCP	66	64784 → 443 [SYN] Seq=40026
11	2019-10-23 07:45:06.216631	192.168.0.100	10.229.20.96	TCP	66	443 → 64784 [SYN, ACK] Seq=
12	2019-10-23 07:45:06.218550	10.229.20.96	192.168.0.100	TCP	60	64784 → 443 [ACK] Seq=40026
13	2019-10-23 07:45:06.219386	10.229.20.96	192.168.0.100	TLSv1.2	571	Client Hello

Tip: gebruik het weergavefilter `tcp.flags==0x2` op Wireshark om te controleren op alle nieuwe TCP-sessies die zijn opgenomen. Dit filtert alle TCP/SYN-pakketten die zijn opgenomen.

No.	Time	Source	Destination	Protocol	Length	Info
10	2019-10-23 07:45:06.216584	10.229.20.96	192.168.0.100	TCP	66	64784 → 443 [SYN] Seq=4002690284 Win=64240 Len=0 MSS=
19	2019-10-23 07:45:06.225743	10.229.20.96	192.168.0.100	TCP	66	64785 → 443 [SYN] Seq=3970528579 Win=64240 Len=0 MSS=
45	2019-10-23 07:45:12.403280	10.229.20.96	192.168.0.100	TCP	66	64790 → 443 [SYN] Seq=442965162 Win=64240 Len=0 MSS=1
51	2019-10-23 07:45:12.409842	10.229.20.96	192.168.0.100	TCP	66	64791 → 443 [SYN] Seq=77539654 Win=64240 Len=0 MSS=13
72	2019-10-23 07:45:14.466836	192.168.0.100	72.163.4.38	TCP	74	35752 → 443 [SYN] Seq=2427943531 Win=29200 Len=0 MSS=
108	2019-10-23 07:45:24.969622	192.168.0.100	72.163.4.38	TCP	74	35756 → 443 [SYN] Seq=1993860949 Win=29200 Len=0 MSS=
137	2019-10-23 07:45:35.469403	192.168.0.100	173.37.145.8	TCP	74	58326 → 443 [SYN] Seq=723413997 Win=29200 Len=0 MSS=1
163	2019-10-23 07:45:45.969384	192.168.0.100	173.37.145.8	TCP	74	58330 → 443 [SYN] Seq=2299582550 Win=29200 Len=0 MSS=
192	2019-10-23 07:45:56.468604	192.168.0.100	72.163.4.38	TCP	74	35768 → 443 [SYN] Seq=1199682453 Win=29200 Len=0 MSS=
227	2019-10-23 07:46:07.218984	10.229.20.96	192.168.0.100	TCP	66	64811 → 443 [SYN] Seq=1496581075 Win=64240 Len=0 MSS=
236	2019-10-23 07:46:07.225881	10.229.20.96	192.168.0.100	TCP	66	64812 → 443 [SYN] Seq=563292608 Win=64240 Len=0 MSS=1

Tip: Pas het veld **Servernaam** toe als kolom vanuit de SSL-client Hallo.



Tip: Pas dit weergavefilter toe om alleen de berichten van Client Hello `ssl.handshake.type == 1` te zien

No.	Time	Source	Destination	Protocol	Length	Server Name	Info
13	2019-10-23 07:45:06.219386	10.229.20.96	192.168.0.100	TLSv1.2	571		Client Hello
23	2019-10-23 07:45:06.227250	10.229.20.96	192.168.0.100	TLSv1.2	571		Client Hello
48	2019-10-23 07:45:12.406366	10.229.20.96	192.168.0.100	TLSv1.2	571		Client Hello
54	2019-10-23 07:45:12.412199	10.229.20.96	192.168.0.100	TLSv1.2	571		Client Hello
75	2019-10-23 07:45:14.634091	192.168.0.100	72.163.4.38	TLSv1.2	571	tools.cisco.com	Client Hello
111	2019-10-23 07:45:25.136089	192.168.0.100	72.163.4.38	TLSv1.2	571	tools.cisco.com	Client Hello
140	2019-10-23 07:45:35.637252	192.168.0.100	173.37.145.8	TLSv1.2	571	tools.cisco.com	Client Hello
166	2019-10-23 07:45:46.136858	192.168.0.100	173.37.145.8	TLSv1.2	571	tools.cisco.com	Client Hello
195	2019-10-23 07:45:56.635438	192.168.0.100	72.163.4.38	TLSv1.2	571	tools.cisco.com	Client Hello
230	2019-10-23 07:46:07.221567	10.229.20.96	192.168.0.100	TLSv1.2	571		Client Hello
240	2019-10-23 07:46:07.228486	10.229.20.96	192.168.0.100	TLSv1.2	571		Client Hello

Opmerking: ten tijde van dit schrijven maakt het Smart Licensing-portal (tools.cisco.com) gebruik van deze IP's: 72.163.4.38, 173.37.145.8

Volg een van de TCP-stromen (**Volgen > TCP-stream**), zoals in de afbeelding.

75	2019-10-23 07:45:14.634091	192.168.0.100	72.163.4.38	TLSv1.2	571	tools.cisco.cc	
111	2019-10-23 07:45:25.136089	192.168.0.100	72.163.4.38	TLSv1.2	571	tools.cisco.cc	
140	2019-10-23 07:45:35.637252	192.168.0.100	173.37.145.8	TLSv1.2	571	tools.cisco.cc	
166	2019-10-23 07:45:46.136858	192.168.0.100	173.37.145.8	TLSv1.2	571	tools.cisco.cc	
195	2019-10-23 07:45:56.635438	192.168.0.100	72.163.4.38	TLSv1.2	571	tools.cisco.cc	
230	2019-10-23 07:46:07.221567	10.229.20.96	192.168.0.100	TLSv1.2	571		
240	2019-10-23 07:46:07.228486	10.229.20.96	192.168.0.100	TLSv1.2	571		

name 75: 571 bytes on wire (4568 bits), 571 bytes captured (4568 bits)
 ethernet II, Src: Vmware_10:d0:a7 (00:0c:29:10:d0:a7), Dst: Cisco_f6:1d:ae (00:be:75:f6:1d:ae)
 Internet Protocol Version 4, Src: 192.168.0.100, Dst: 72.163.4.38
 Transmission Control Protocol, Src Port: 35752, Dst Port: 443, Seq: 2427943532, Ack: 2770078885, Len: 571
 Secure Sockets Layer

▼ TLSv1.2 Record Layer: Handshake Protocol: Client Hello
 Content Type: Handshake (22)
 Version: TLS 1.0 (0x0301)
 Length: 512

- Mark/Unmark Packet
- Ignore/Unignore Packet
- Set/Unset Time Reference
- Time Shift...
- Packet Comment...
- Edit Resolved Name
- Apply as Filter
- Prepare a Filter
- Conversation Filter
- Colorize Conversation
- SCTP
- Follow
- Copy
- Protocol Preferences
- Decode As...
- Show Packet in New Window

No.	Time	Source	Destination	Protocol	Length	Server Name	Info
72	2019-10-23 07:45:14.466836	192.168.0.100	72.163.4.38	TCP	74		35752 → 443 [SYN] Seq=2427943531 Win=29200 Len=0 MSS=
73	2019-10-23 07:45:14.632885	72.163.4.38	192.168.0.100	TCP	60		443 → 35752 [SYN, ACK] Seq=2770078884 Ack=2427943532
74	2019-10-23 07:45:14.632935	192.168.0.100	72.163.4.38	TCP	54		35752 → 443 [ACK] Seq=2427943532 Ack=2770078885 Win=
75	2019-10-23 07:45:14.634091	192.168.0.100	72.163.4.38	TLSv1.2	571	tools.cisco.com	Client Hello
76	2019-10-23 07:45:14.634796	72.163.4.38	192.168.0.100	TCP	60		443 → 35752 [ACK] Seq=2770078885 Ack=2427944049 Win=
77	2019-10-23 07:45:14.966729	72.163.4.38	192.168.0.100	TLSv1.2	150		Server Hello
78	2019-10-23 07:45:14.966772	192.168.0.100	72.163.4.38	TCP	54		35752 → 443 [ACK] Seq=2427944049 Ack=2770078981 Win=
79	2019-10-23 07:45:14.966834	72.163.4.38	192.168.0.100	TCP	1384		443 → 35752 [PSH, ACK] Seq=2770078981 Ack=2427944049
80	2019-10-23 07:45:14.966880	192.168.0.100	72.163.4.38	TCP	54		35752 → 443 [ACK] Seq=2427944049 Ack=2770080311 Win=
81	2019-10-23 07:45:14.966877	72.163.4.38	192.168.0.100	TLSv1.2	155		Certificate
82	2019-10-23 07:45:14.966880	192.168.0.100	72.163.4.38	TCP	54		35752 → 443 [ACK] Seq=2427944049 Ack=2770080412 Win=
83	2019-10-23 07:45:14.966915	72.163.4.38	192.168.0.100	TLSv1.2	63		Server Hello Done
84	2019-10-23 07:45:14.966925	192.168.0.100	72.163.4.38	TCP	54		35752 → 443 [ACK] Seq=2427944049 Ack=2770080421 Win=
85	2019-10-23 07:45:14.967114	192.168.0.100	72.163.4.38	TLSv1.2	61		Alert (Level: Fatal, Description: Unknown CA)
86	2019-10-23 07:45:14.967261	192.168.0.100	72.163.4.38	TCP	54		35752 → 443 [RST, ACK] Seq=2427944056 Ack=2770080421 Win=
87	2019-10-23 07:45:14.967382	72.163.4.38	192.168.0.100	TCP	60		443 → 35752 [ACK] Seq=2770080421 Ack=2427944056 Win=
88	2019-10-23 07:45:14.967398	192.168.0.100	72.163.4.38	TCP	54		35752 → 443 [RST] Seq=2427944056 Win=0 Len=0

> Frame 75: 571 bytes on wire (4568 bits), 571 bytes captured (4568 bits)
 > Ethernet II, Src: Vmware_10:d0:a7 (00:0c:29:10:d0:a7), Dst: Cisco_f6:1d:ae (00:be:75:f6:1d:ae)
 > Internet Protocol Version 4, Src: 192.168.0.100, Dst: 72.163.4.38
 > Transmission Control Protocol, Src Port: 35752, Dst Port: 443, Seq: 2427943532, Ack: 2770078885, Len: 517
 ▼ Secure Sockets Layer

▼ TLSv1.2 Record Layer: Handshake Protocol: Client Hello
 Content Type: Handshake (22)
 Version: TLS 1.0 (0x0301)
 Length: 512

▼ Handshake Protocol: Client Hello
 Handshake Type: Client Hello (1)
 Length: 508
 Version: TLS 1.2 (0x0303)
 > Random: 234490a107438c73b59564653271c7c09fbbb7ac16897184...
 Session ID Length: 0
 Cipher Suites Length: 100
 > Cipher Suites (50 suites)

Belangrijkste punten:

1. Er is een TCP 3-weg handdruk.
2. De client (FMC) stuurt een SSL-client-Hello-bericht naar het Smart Licensing-portal.
3. De SSL-sessie-id is 0. Dit betekent dat het geen hervatte zitting is.
4. De doelserver antwoordt met Server Hello, Certificaat en Server Hello done bericht.
5. De klant stuurt een SSL Fatal Alert met betrekking tot een "Onbekende CA".
6. De client stuurt een TCP/RST om de sessie te sluiten.
7. De gehele duur van de TCP-sessie (van vestiging tot sluiting) was ~0,5 sec.

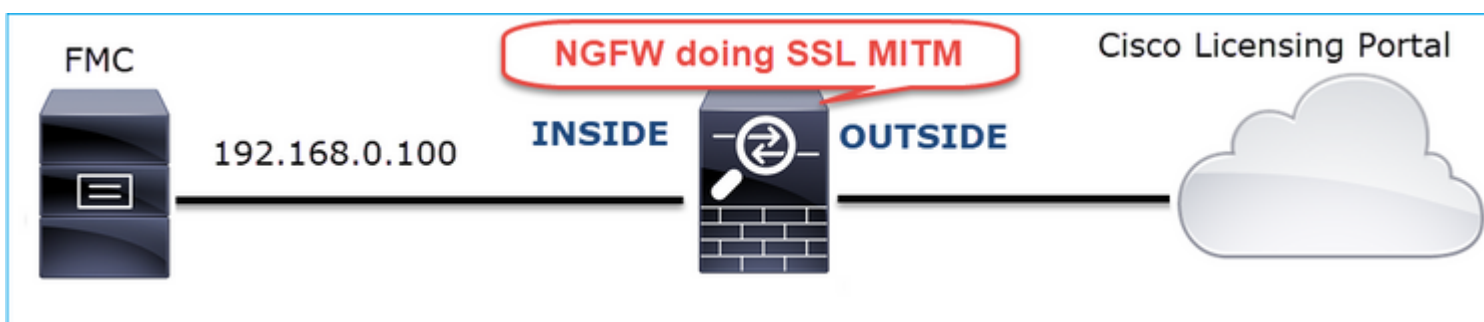
Selecteer het **Servercertificaat** en vouw het veld **voor** de **emittent** uit om de algemene naam te zien. In dit geval toont de algemene naam een apparaat dat Man-in-the-middle (MITM) doet.

No.	Time	Source	Destination	Protocol	Length	Server Name	Info
72	2019-10-23 07:45:14.466836	192.168.0.100	72.163.4.38	TCP	74		35752 → 443 [SYN] Seq=2427943531 Wi
73	2019-10-23 07:45:14.632885	72.163.4.38	192.168.0.100	TCP	60		443 → 35752 [SYN, ACK] Seq=27700788
74	2019-10-23 07:45:14.632935	192.168.0.100	72.163.4.38	TCP	54		35752 → 443 [ACK] Seq=2427943532 Ac
75	2019-10-23 07:45:14.634091	192.168.0.100	72.163.4.38	TLSv1.2	571	tools.cisco.com	Client Hello
76	2019-10-23 07:45:14.634796	72.163.4.38	192.168.0.100	TCP	60		443 → 35752 [ACK] Seq=2770078885 Ac
77	2019-10-23 07:45:14.966729	72.163.4.38	192.168.0.100	TLSv1.2	150		Server Hello
78	2019-10-23 07:45:14.966772	192.168.0.100	72.163.4.38	TCP	54		35752 → 443 [ACK] Seq=2427944049 Ac
79	2019-10-23 07:45:14.966834	72.163.4.38	192.168.0.100	TCP	1384		443 → 35752 [PSH, ACK] Seq=27700789
80	2019-10-23 07:45:14.966850	192.168.0.100	72.163.4.38	TCP	54		35752 → 443 [ACK] Seq=2427944049 Ac
81	2019-10-23 07:45:14.966872	72.163.4.38	192.168.0.100	TLSv1.2	155		Certificate

```

Length: 1426
  Handshake Protocol: Certificate
    Handshake Type: Certificate (11)
    Length: 1422
    Certificates Length: 1419
  Certificates (1419 bytes)
    Certificate Length: 1416
  Certificate: 308205843082046ca003020102020d00aa23af5d607e0000... (id-at-commonName=tools.cisco.com,id-at-organizationName=Cisco Sy
    signedCertificate
      version: v3 (2)
      serialNumber: 0x00aa23af5d607e00002f423880
      signature (sha256WithRSAEncryption)
        issuer: rdnSequence (0)
          rdnSequence: 3 items (id-at-commonName=FTD4100_MITM,id-at-organizationalUnitName=FTD_OU,id-at-organizationName=FTD_O)
            RDNSquence item: 1 item (id-at-organizationName=FTD_O)
            RDNSquence item: 1 item (id-at-organizationalUnitName=FTD_OU)
            RDNSquence item: 1 item (id-at-commonName=FTD4100_MITM)
          validity
          subject: rdnSequence (0)
          subjectPublicKeyInfo
        extensions: 6 items
  
```

Dit wordt in deze afbeelding getoond:

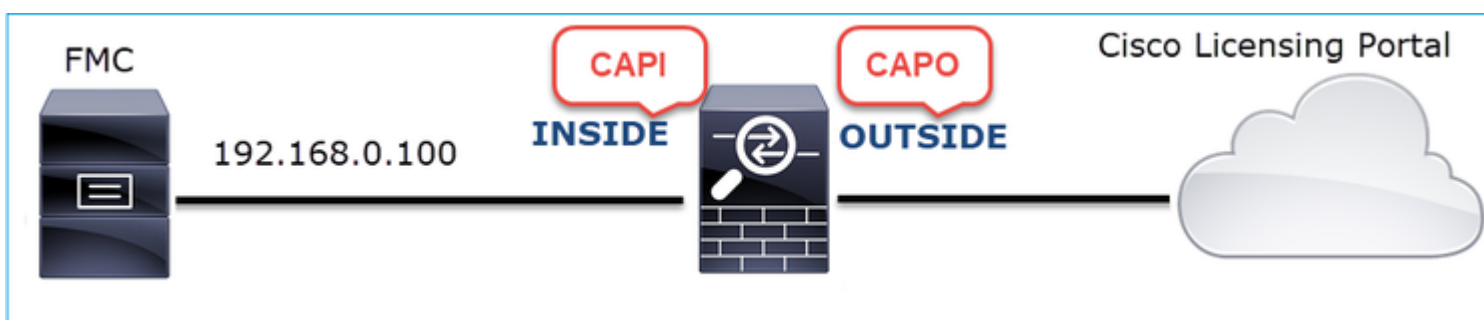


Aanbevolen acties

De acties die in deze paragraaf worden opgesomd, hebben tot doel de kwestie verder af te zwakken.

Actie 1. Neem extra opnamen.

Leg opnamen vast op het transitfirewall-apparaat:



CAPI laat zien:

tcp.stream eq 57

No.	Time	Source	Destination	Protocol	Length	Server Name	Info
1221	2019-10-22 17:49:03.212681	192.168.0.100	173.37.145.8	TCP	74		39924 → 443 [SYN] Seq=42
1222	2019-10-22 17:49:03.379023	173.37.145.8	192.168.0.100	TCP	58		443 → 39924 [SYN, ACK] S
1223	2019-10-22 17:49:03.379298	192.168.0.100	173.37.145.8	TCP	54		39924 → 443 [ACK] Seq=42
1224	2019-10-22 17:49:03.380336	192.168.0.100	173.37.145.8	TLSv1.2	571	tools.cisco.com	Client Hello
1225	2019-10-22 17:49:03.380732	173.37.145.8	192.168.0.100	TCP	54		443 → 39924 [ACK] Seq=23
1226	2019-10-22 17:49:03.710092	173.37.145.8	192.168.0.100	TLSv1.2	150		Server Hello
1227	2019-10-22 17:49:03.710092	173.37.145.8	192.168.0.100	TCP	1384		443 → 39924 [PSH, ACK] S
1228	2019-10-22 17:49:03.710092	173.37.145.8	192.168.0.100	TLSv1.2	155		Certificate
1229	2019-10-22 17:49:03.710107	173.37.145.8	192.168.0.100	TLSv1.2	63		Server Hello Done
1230	2019-10-22 17:49:03.710412	192.168.0.100	173.37.145.8	TCP	54		39924 → 443 [ACK] Seq=42
1231	2019-10-22 17:49:03.710519	192.168.0.100	173.37.145.8	TCP	54		39924 → 443 [ACK] Seq=42
1232	2019-10-22 17:49:03.710519	192.168.0.100	173.37.145.8	TCP	54		39924 → 443 [ACK] Seq=42
1233	2019-10-22 17:49:03.710534	192.168.0.100	173.37.145.8	TCP	54		39924 → 443 [ACK] Seq=42
1234	2019-10-22 17:49:03.710626	192.168.0.100	173.37.145.8	TLSv1.2	61		Alert (Level: Fatal, Des
1235	2019-10-22 17:49:03.710641	173.37.145.8	192.168.0.100	TCP	54		443 → 39924 [ACK] Seq=23
1236	2019-10-22 17:49:03.710748	192.168.0.100	173.37.145.8	TCP	54		39924 → 443 [RST, ACK] S
1237	2019-10-22 17:49:03.710870	192.168.0.100	173.37.145.8	TCP	54		39924 → 443 [RST] Seq=42

<

Length: 1426

- Handshake Protocol: Certificate
 - Handshake Type: Certificate (11)
 - Length: 1422
 - Certificates Length: 1419
- Certificates (1419 bytes)
 - Certificate Length: 1416
 - Certificate: 308205843082046ca003020102020d00aa23af5d607e0000... (id-at-commonName=tools.cisco.com,id-at-organizationName=)
 - signedCertificate
 - version: v3 (2)
 - serialNumber: 0x00aa23af5d607e00002f423880
 - signature (sha256WithRSAEncryption)
 - issuer: rdnSequence (0)
 - rdnSequence: 3 items (id-at-commonName=FTD4100_MITM,id-at-organizationalUnitName=FTD_OU,id-at-organizationName=)
 - RDNSquence item: 1 item (id-at-organizationName=FTD_O)
 - RDNSquence item: 1 item (id-at-organizationalUnitName=FTD_OU)
 - RDNSquence item: 1 item (id-at-commonName=FTD4100_MITM)
 - validity

CAPO laat zien:

No.	Time	Source	Destination	Protocol	Length	Server Name	Info
1169	2019-10-22 17:49:03.212849	192.168.0.100	173.37.145.8	TCP	78		39924 → 443 [SYN] Seq=623942
1170	2019-10-22 17:49:03.378962	173.37.145.8	192.168.0.100	TCP	62		443 → 39924 [SYN, ACK] Seq=4
1171	2019-10-22 17:49:03.379329	192.168.0.100	173.37.145.8	TCP	58		39924 → 443 [ACK] Seq=623942
1172	2019-10-22 17:49:03.380793	192.168.0.100	173.37.145.8	TLSv1.2	512	tools.cisco.com	Client Hello
1173	2019-10-22 17:49:03.545748	173.37.145.8	192.168.0.100	TCP	1388		443 → 39924 [PSH, ACK] Seq=4
1174	2019-10-22 17:49:03.545809	173.37.145.8	192.168.0.100	TCP	1388		443 → 39924 [PSH, ACK] Seq=4
1175	2019-10-22 17:49:03.545824	192.168.0.100	173.37.145.8	TCP	58		39924 → 443 [ACK] Seq=623942
1176	2019-10-22 17:49:03.545915	173.37.145.8	192.168.0.100	TCP	1388		443 → 39924 [PSH, ACK] Seq=4
1177	2019-10-22 17:49:03.545961	173.37.145.8	192.168.0.100	TCP	1388		443 → 39924 [PSH, ACK] Seq=4
1178	2019-10-22 17:49:03.545961	192.168.0.100	173.37.145.8	TCP	58		39924 → 443 [ACK] Seq=623942
1179	2019-10-22 17:49:03.709420	173.37.145.8	192.168.0.100	TLSv1.2	82		Server Hello, Certificate, S
1180	2019-10-22 17:49:03.710687	192.168.0.100	173.37.145.8	TLSv1.2	65		Alert (Level: Fatal, Descrip
1181	2019-10-22 17:49:03.710885	192.168.0.100	173.37.145.8	TCP	58		39924 → 443 [FIN, PSH, ACK]
1182	2019-10-22 17:49:03.874542	173.37.145.8	192.168.0.100	TCP	58		443 → 39924 [RST, ACK] Seq=4


```

Length: 5339
> Handshake Protocol: Server Hello
< Handshake Protocol: Certificate
  Handshake Type: Certificate (11)
  Length: 5240
  Certificates Length: 5237
  < Certificates (5237 bytes)
    Certificate Length: 2025
    < Certificate: 308207e5308205cda00302010202143000683b0f7504f7b2... (id-at-commonName=tools.cisco.com,id-at-organizationName=C
      > signedCertificate
      > algorithmIdentifier (sha256WithRSAEncryption)
      Padding: 0
      encrypted: 6921d084f7a6f6167058f14e2aad8b98b4e6c971ea6ea3b4...
    Certificate Length: 1736
    < Certificate: 308206c4308204aca00302010202147517167783d0437eb5... (id-at-commonName=HydrantID SSL ICA G2,id-at-organizationName=
      < signedCertificate
        version: v3 (2)
        serialNumber: 0x7517167783d0437eb556c357946e4563b8ebd3ac
      > signature (sha256WithRSAEncryption)
      < issuer: rdnSequence (0)
        > rdnSequence: 3 items (id-at-commonName=QuoVadis Root CA 2,id-at-organizationName=QuoVadis Limited,id-at-countryName=US)
      > validity
  
```

Hiermee wordt aangetoond dat de transitfirewall het servercertificaat (MITM) wijzigt

Actie 2. Controleer de apparaatlogboeken.

U kunt de FMC TS-bundel verzamelen zoals beschreven in dit document:

<https://www.cisco.com/c/en/us/support/docs/security/sourcefire-defense-center/117663-technote-SourceFire-00.html>

In dit geval toont het `/dir-archives/var-log/process_stdout.log`-bestand berichten als dit:

```

<#root>
SOUT: 10-23 05:45:14 2019-10-23 05:45:36 sla[10068]: *Wed .967 UTC: CH-LIB-ERROR: ch_pf_curl_send_msg[49]
failed to perform, err code 60, err string "SSL peer certificate or SSH remote key was not OK"
...
SOUT: 10-23 05:45:14 2019-10-23 05:45:36 sla[10068]: *Wed .967 UTC: CH-LIB-TRACE: ch_pf_curl_is_cert_iss
cert issue checking, ret 60, url "https://tools.cisco.com/its/

```

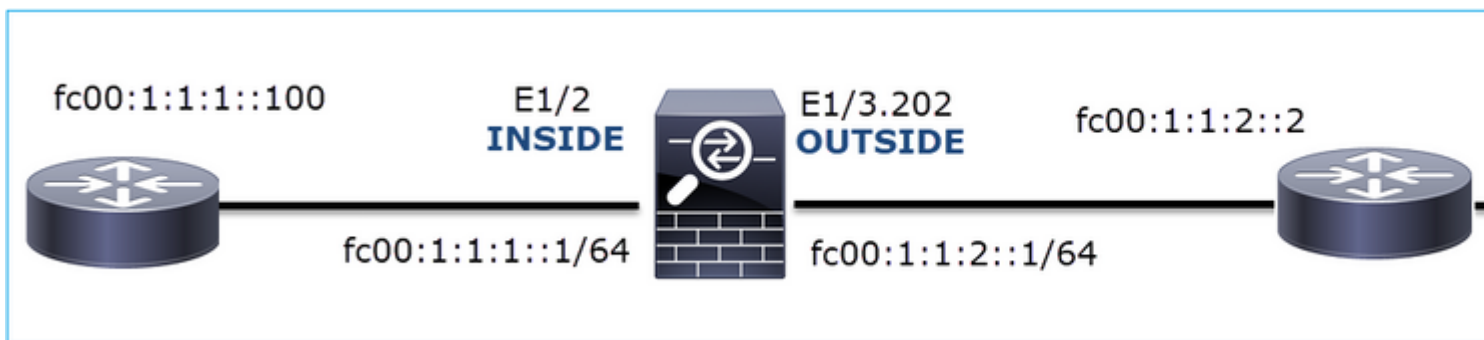
Aanbevolen oplossing

Schakel de MITM uit voor de specifieke stroom, zodat het VCC zich met succes kan registreren in de Smart Licensing-cloud.

Situatie 11. IPv6-connectiviteitsprobleem

Probleem Beschrijving: Interne hosts (die zich achter de BINNENKANT-interface van de firewall bevinden) kunnen niet communiceren met externe hosts (hosts die zich achter de BUITEN-interface van de firewall bevinden).

Dit beeld toont de topologie:



Beïnvloede stroom:

SRC IP: `fc00:1:1:1:10`

Gesch. IP: `fc00:1:1:2:2`

Protocol: alle

Capture Analysis

Schakel opnamen in op de FTD LINA-motor.

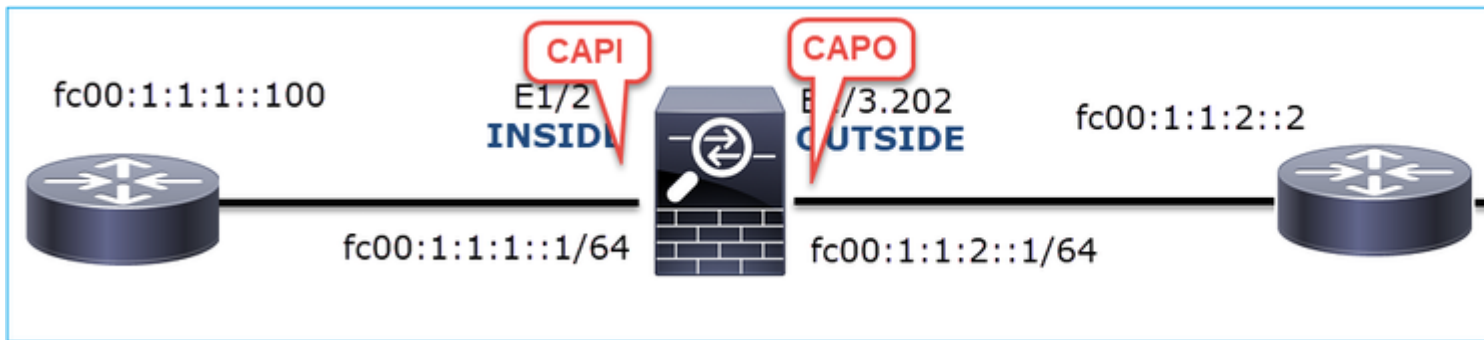
```
<#root>
```

```
firepower#
```

```
capture CAPI int INSIDE match ip any6 any6
```

```
firepower#
```

```
capture CAPO int OUTSIDE match ip any6 any6
```



Captures - niet-functioneel scenario

Deze opnamen werden genomen parallel met een ICMP-connectiviteitstest van IP fc00:1:1:1:100 (interne router) naar IP fc00:1:1:2:2 (upstream router).

De opname op firewall INSIDE interface bevat:

No.	Time	Source	Destination	Protocol	Length	Info
1	2019-10-24 13:02:07.001663	fc00:1:1:1::100	ff02::1:ff00:1	ICMPv6	86	Neighbor Solicitation for fc00:1:1:1::1
2	2019-10-24 13:02:07.001876	fc00:1:1:1::1	fc00:1:1:1::100	ICMPv6	86	Neighbor Advertisement fc00:1:1:1::1
3	2019-10-24 13:02:07.002273	fc00:1:1:1::100	fc00:1:1:2::2	ICMPv6	114	Echo (ping) request id=0x160d
4	2019-10-24 13:02:08.997918	fc00:1:1:1::100	fc00:1:1:2::2	ICMPv6	114	Echo (ping) request id=0x160d
5	2019-10-24 13:02:10.998056	fc00:1:1:1::100	fc00:1:1:2::2	ICMPv6	114	Echo (ping) request id=0x160d
6	2019-10-24 13:02:11.999917	fe80::2be:75ff:fef6:1dae	fc00:1:1:1::100	ICMPv6	86	Neighbor Solicitation for fc00:1:1:1::1
7	2019-10-24 13:02:12.002075	fc00:1:1:1::100	fe80::2be:75ff:fef6:1dae	ICMPv6	78	Neighbor Advertisement fc00:1:1:1::1
8	2019-10-24 13:02:12.998346	fc00:1:1:1::100	fc00:1:1:2::2	ICMPv6	114	Echo (ping) request id=0x160d
9	2019-10-24 13:02:14.998483	fc00:1:1:1::100	fc00:1:1:2::2	ICMPv6	114	Echo (ping) request id=0x160d
10	2019-10-24 13:02:17.062725	fe80::4e4e:35ff:fefc:fcd8	fe80::2be:75ff:fef6:1dae	ICMPv6	86	Neighbor Solicitation for fe80::2be:75ff:fef6:1dae
11	2019-10-24 13:02:17.062862	fe80::2be:75ff:fef6:1dae	fe80::4e4e:35ff:fefc:fcd8	ICMPv6	78	Neighbor Advertisement fe80::2be:75ff:fef6:1dae
12	2019-10-24 13:02:22.059994	fe80::2be:75ff:fef6:1dae	fe80::4e4e:35ff:fefc:fcd8	ICMPv6	86	Neighbor Solicitation for fe80::4e4e:35ff:fefc:fcd8
13	2019-10-24 13:02:22.063000	fe80::4e4e:35ff:fefc:fcd8	fe80::2be:75ff:fef6:1dae	ICMPv6	78	Neighbor Advertisement fe80::4e4e:35ff:fefc:fcd8

Belangrijkste punten:

1. De router verstuurt een IPv6 Neighbor Solicitation-bericht en vraagt om het MAC-adres van het upstream-apparaat (IP fc00:1:1:1:1).
2. De firewall reageert met een IPv6 Neighbor Advertisement.
3. De router verzendt een ICMP-echo-verzoek.
4. De firewall verstuurt een IPv6 Neighbor Solicitation-bericht en vraagt om het MAC-adres van het downstream apparaat (fc00:1:1:100).
5. De router antwoordt met een IPv6-burenadvertentie.
6. De router verstuurt extra IPv6 ICMP-echoaanvragen.

De opname op de firewall BUITEN interface bevat:

No.	Time	Source	Destination	Protocol	Length	Info
1	2019-10-24 13:02:07.002517	fe80::2be:75ff:fef6:1d8e	ff02::1:ff00:2	ICMPv6	90	Neighbor Solicitation for fc00:1:1:2::2
2	2019-10-24 13:02:07.005569	fc00:1:1:2::2	fe80::2be:75ff:fef6:1d8e	ICMPv6	90	Neighbor Advertisement fc00:1:1:2::2
3	2019-10-24 13:02:08.997995	fc00:1:1:1::100	fc00:1:1:2::2	ICMPv6	18	Echo (ping) request id=0x160d
4	2019-10-24 13:02:09.001815	fc00:1:1:2::2	ff02::1:ff00:100	ICMPv6	90	Neighbor Solicitation for fc00:1:1:2::2
5	2019-10-24 13:02:10.025938	fc00:1:1:2::2	ff02::1:ff00:100	ICMPv6	90	Neighbor Solicitation for fc00:1:1:2::2
6	2019-10-24 13:02:10.998132	fc00:1:1:1::100	fc00:1:1:2::2	ICMPv6	118	Echo (ping) request id=0x160d
7	2019-10-24 13:02:11.050015	fc00:1:1:2::2	ff02::1:ff00:100	ICMPv6	90	Neighbor Solicitation for fc00:1:1:2::2
8	2019-10-24 13:02:12.066082	fe80::4e4e:35ff:fefc:fcd8	fe80::2be:75ff:fef6:1d8e	ICMPv6	90	Neighbor Solicitation for fe80::2be:75ff:fef6:1d8e
9	2019-10-24 13:02:12.066234	fe80::2be:75ff:fef6:1d8e	fe80::4e4e:35ff:fefc:fcd8	ICMPv6	82	Neighbor Advertisement fe80::2be:75ff:fef6:1d8e
10	2019-10-24 13:02:12.998422	fc00:1:1:1::100	fc00:1:1:2::2	ICMPv6	118	Echo (ping) request id=0x160d
11	2019-10-24 13:02:13.002105	fc00:1:1:2::2	ff02::1:ff00:100	ICMPv6	90	Neighbor Solicitation for fc00:1:1:2::2
12	2019-10-24 13:02:14.090251	fc00:1:1:2::2	ff02::1:ff00:100	ICMPv6	90	Neighbor Solicitation for fc00:1:1:2::2
13	2019-10-24 13:02:14.998544	fc00:1:1:1::100	fc00:1:1:2::2	ICMPv6	118	Echo (ping) request id=0x160d
14	2019-10-24 13:02:15.178350	fc00:1:1:2::2	ff02::1:ff00:100	ICMPv6	90	Neighbor Solicitation for fc00:1:1:2::2
15	2019-10-24 13:02:17.059963	fe80::2be:75ff:fef6:1d8e	fe80::4e4e:35ff:fefc:fcd8	ICMPv6	90	Neighbor Solicitation for fe80::2be:75ff:fef6:1d8e
16	2019-10-24 13:02:17.062512	fe80::4e4e:35ff:fefc:fcd8	fe80::2be:75ff:fef6:1d8e	ICMPv6	82	Neighbor Advertisement fe80::2be:75ff:fef6:1d8e

Belangrijkste punten:

1. De firewall verstuurt een IPv6 Neighbor Solicitation-bericht waarin wordt gevraagd naar het MAC-adres van het upstream-apparaat (IP fc00:1:1:2:2).
2. De router antwoordt met een IPv6-burenadvertentie.
3. De firewall verzendt een IPv6 ICMP-echo-verzoek.
4. Het stroomopwaartse apparaat (router fc00:1:1:2:2) verstuurt een IPv6 buurvraag bericht dat vraagt om het MAC-adres van het IPv6 adres fc00:1:1:1:100.
5. De firewall verzendt een extra IPv6 ICMP-echo-verzoek.
6. De upstream router verstuurt een extra IPv6 Neighbor Solicitation-bericht waarin wordt gevraagd naar het MAC-adres van het IPv6-adres fc00:1:1:1:100.

Punt 4 is zeer interessant. Normaal vraagt de upstream router om de MAC van de firewall BUITEN interface (fc00:1:1:2:2), maar in plaats daarvan vraagt hij om de fc00:1:1:1:100. Dit is een indicatie van een verkeerde configuratie.

Aanbevolen acties

De acties die in deze paragraaf worden opgesomd, hebben tot doel de kwestie verder af te zwakken.

Actie 1. Controleer de IPv6-buurtabel.

De IPv6-burentabel voor firewalls is correct ingevuld.

```
<#root>
```

```
firepower#
```

```
show ipv6 neighbor | i fc00
```

```
fc00:1:1:2::2          58 4c4e.35fc.fcd8  STALE OUTSIDE
fc00:1:1:1::100       58 4c4e.35fc.fcd8  STALE INSIDE
```

Actie 2. Controleer de IPv6-configuratie.

Dit is de firewallconfiguratie.

```
<#root>
```

```
firewall#
```

```
show run int e1/2
```

```
!
interface Ethernet1/2
 nameif INSIDE
 cts manual
 propagate sgt preserve-untag
 policy static sgt disabled trusted
 security-level 0
 ip address 192.168.0.1 255.255.255.0
 ipv6 address
```

```
fc00:1:1:1::1/64
```

```
ipv6 enable
```

```

firewall#
show run int e1/3.202
!
interface Ethernet1/3.202
vlan 202
nameif OUTSIDE
cts manual
propagate sgt preserve-untag
policy static sgt disabled trusted
security-level 0
ip address 192.168.103.96 255.255.255.0
ipv6 address
fc00:1:1:2::1/64

ipv6 enable

```

De stroomopwaartse apparatenconfiguratie openbaart de misconfiguratie:

```

<#root>
Router#
show run interface g0/0.202
!
interface GigabitEthernet0/0.202
encapsulation dot1Q 202
vrf forwarding VRF202
ip address 192.168.2.72 255.255.255.0
ipv6 address FC00:1:1:2::2
/48

```

Opname - functioneel scenario

De subnetmasker verandering (van /48 naar /64) heeft het probleem opgelost. Dit is de CAPI-opname in het functionele scenario.

No.	Time	Source	Destination	Protocol	Length	Info
1	2019-10-24 15:17:20.677775	fc00:1:1:1::100	ff02::1:ff00:1	ICMPv6	86	Neighbor Solicitation for fc00:1:1:2::2
2	2019-10-24 15:17:20.677989	fc00:1:1:1::1	fc00:1:1:1::100	ICMPv6	86	Neighbor Advertisement fc00:1:1:2::2
3	2019-10-24 15:17:20.678401	fc00:1:1:1::100	fc00:1:1:2::2	ICMPv6	114	Echo (ping) request id=0x097e, seq=1
4	2019-10-24 15:17:22.674281	fc00:1:1:1::100	fc00:1:1:2::2	ICMPv6	114	Echo (ping) request id=0x097e, seq=2
5	2019-10-24 15:17:24.674403	fc00:1:1:1::100	fc00:1:1:2::2	ICMPv6	114	Echo (ping) request id=0x097e, seq=3
6	2019-10-24 15:17:24.674815	fc00:1:1:2::2	fc00:1:1:1::100	ICMPv6	114	Echo (ping) reply id=0x097e, seq=1
7	2019-10-24 15:17:24.675242	fc00:1:1:1::100	fc00:1:1:2::2	ICMPv6	114	Echo (ping) request id=0x097e, seq=4
8	2019-10-24 15:17:24.675731	fc00:1:1:2::2	fc00:1:1:1::100	ICMPv6	114	Echo (ping) reply id=0x097e, seq=2
9	2019-10-24 15:17:24.676356	fc00:1:1:1::100	fc00:1:1:2::2	ICMPv6	114	Echo (ping) request id=0x097e, seq=5
10	2019-10-24 15:17:24.676753	fc00:1:1:2::2	fc00:1:1:1::100	ICMPv6	114	Echo (ping) reply id=0x097e, seq=3

Belangrijkste punt:

1. De router verstuurt een IPv6-buurverzoek bericht waarin wordt gevraagd naar het MAC-adres van het

- upstream-apparaat (IP fc00:1:1:1:1).
- 2. De firewall reageert met een IPv6 Neighbor Advertisement.
- 3. De router verstuurt ICMP-echoverzoeken en ontvangt Echo-antwoorden.

CAPO inhoud:

No.	Time	Source	Destination	Protocol	Length	Info
1	2019-10-24 15:17:20.678645	fe80::2be:75ff:fe...	ff02::1:ff00:2	ICMPv6	90	Neighbor Solicitation for fc00:1:1:2::2
2	2019-10-24 15:17:20.681818	fc00:1:1:2::2	fe80::2be:75ff:fe...	ICMPv6	90	Neighbor Advertisement for fc00:1:1:2::2
3	2019-10-24 15:17:22.674342	fc00:1:1:1::100	fc00:1:1:2::2	ICMPv6	118	Echo (ping) request id=0x097e, seq=1
4	2019-10-24 15:17:22.677943	fc00:1:1:2::2	ff02::1:ff00:1	ICMPv6	90	Neighbor Solicitation for fc00:1:1:2::2
5	2019-10-24 15:17:22.678096	fc00:1:1:2::1	fc00:1:1:2::2	ICMPv6	90	Neighbor Advertisement for fc00:1:1:2::1
6	2019-10-24 15:17:22.678462	fc00:1:1:2::2	fc00:1:1:1::100	ICMPv6	118	Echo (ping) reply id=0x097e, seq=1
7	2019-10-24 15:17:24.674449	fc00:1:1:1::100	fc00:1:1:2::2	ICMPv6	118	Echo (ping) request id=0x097e, seq=2
8	2019-10-24 15:17:24.674785	fc00:1:1:2::2	fc00:1:1:1::100	ICMPv6	118	Echo (ping) reply id=0x097e, seq=2
9	2019-10-24 15:17:24.675395	fc00:1:1:1::100	fc00:1:1:2::2	ICMPv6	118	Echo (ping) request id=0x097e, seq=3
10	2019-10-24 15:17:24.675700	fc00:1:1:2::2	fc00:1:1:1::100	ICMPv6	118	Echo (ping) reply id=0x097e, seq=3
11	2019-10-24 15:17:24.676448	fc00:1:1:1::100	fc00:1:1:2::2	ICMPv6	118	Echo (ping) request id=0x097e, seq=4
12	2019-10-24 15:17:24.676738	fc00:1:1:2::2	fc00:1:1:1::100	ICMPv6	118	Echo (ping) reply id=0x097e, seq=4

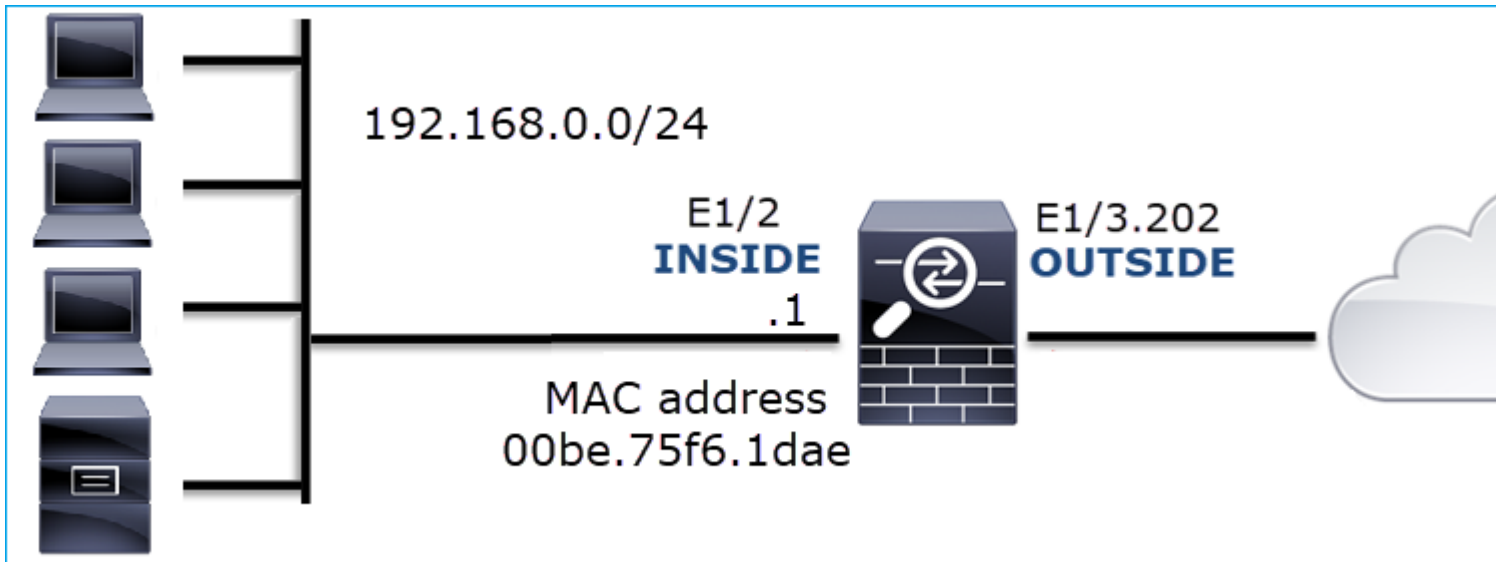
Belangrijkste punten:

1. De firewall verstuurt een IPv6 Neighbor Solicitation-bericht waarin wordt gevraagd naar het MAC-adres van het upstream-apparaat (IP fc00:1:1:2:2).
2. De firewall reageert met een IPv6 Neighbor Advertisement.
3. De firewall verzendt een ICMP-echo-verzoek.
4. De router verstuurt een IPv6 Neighbor Query-bericht waarin wordt gevraagd naar het MAC-adres van het downstream-apparaat (IP fc00:1:1:1:1).
5. De firewall reageert met een IPv6 Neighbor Advertisement.
6. De firewall verzendt ICMP-echoverzoeken en ontvangt Echo-antwoorden.

Situatie 12. Probleem met intermitterende connectiviteit (ARP-vergiftiging)

Probleem Beschrijving: Interne hosts (192.168.0.x/24) hebben intermitterende connectiviteitsproblemen met hosts in dezelfde subnetverbinding

Dit beeld toont de topologie:



Beïnvloede stroom:

SRC IP: 192.168.0.x/24

Dst IP: 192.168.0.x/24

Protocol: alle

De ARP cache van een interne host lijkt vergiftigd te zijn:

```

C:\Windows\system32\cmd.exe
C:\Users\mzafeiro1>arp -a

Interface: 192.168.0.55 --- 0xb
Internet Address      Physical Address      Type
192.168.0.1           00-be-75-f6-1d-ae    dynamic
192.168.0.22          00-be-75-f6-1d-ae    dynamic
192.168.0.23          00-be-75-f6-1d-ae    dynamic
192.168.0.24          00-be-75-f6-1d-ae    dynamic
192.168.0.25          00-be-75-f6-1d-ae    dynamic
192.168.0.26          00-be-75-f6-1d-ae    dynamic
192.168.0.27          00-be-75-f6-1d-ae    dynamic
192.168.0.28          00-be-75-f6-1d-ae    dynamic
192.168.0.29          00-be-75-f6-1d-ae    dynamic
192.168.0.30          00-be-75-f6-1d-ae    dynamic
192.168.0.88          00-be-75-f6-1d-ae    dynamic
192.168.0.255        ff-ff-ff-ff-ff-ff    static
224.0.0.22           01-00-5e-00-00-16    static
224.0.0.251          01-00-5e-00-00-fb    static
224.0.0.252          01-00-5e-00-00-fc    static
239.255.255.250     01-00-5e-7f-ff-fa    static

C:\Users\mzafeiro1>

```

Capture Analysis

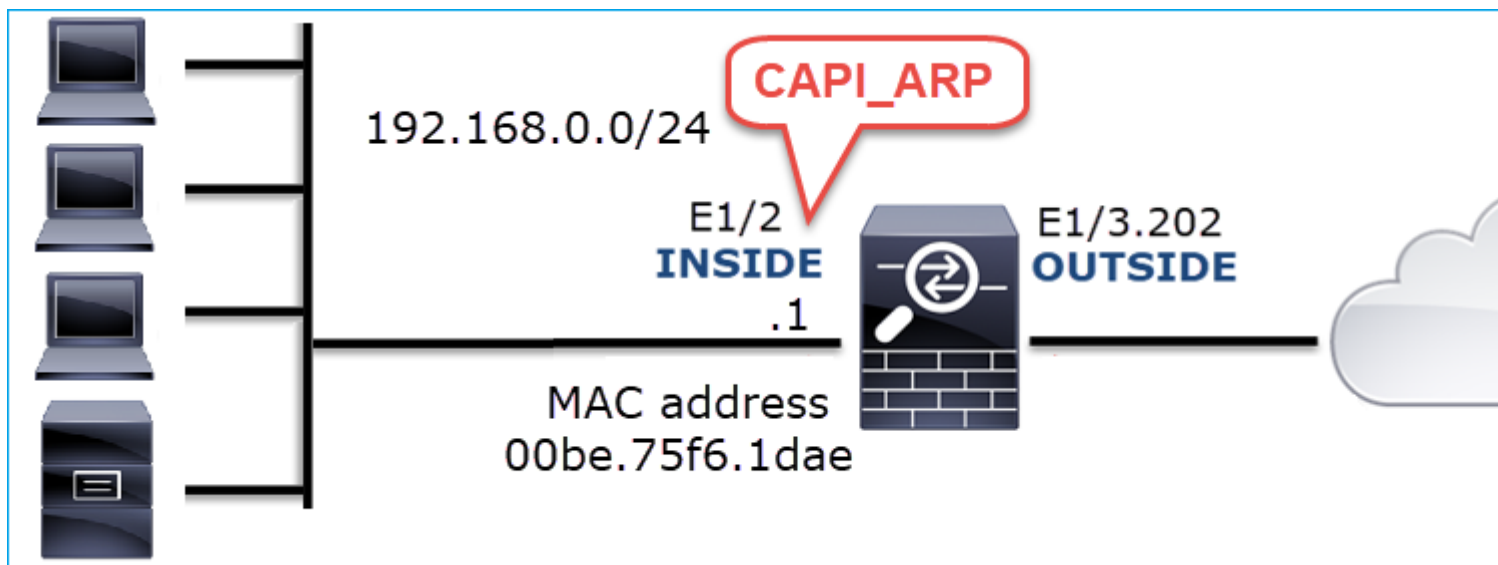
Een opname op FTD LINA-motor inschakelen

Deze opname neemt alleen ARP-pakketten op de BINNENKANT-interface op:

```
<#root>
```

```
firepower#
```

capture CAPI_ARP interface INSIDE ethernet-type arp



Captures - niet-functioneel scenario:

De opname op de firewall INSIDE interface bevat.

No.	Time	Source	Destination	Protocol	Length	Info
4	2019-10-25 10:01:55.179571	Vmware_2c:9b:a7	Broadcast	ARP	1	60 Who has 192.168.0.1
5	2019-10-25 10:01:55.17969	Cisco_f6:1d:ae	Vmware_2c:9b:a7	ARP	1	42 192.168.0.23
35	2019-10-25 10:02:13.050397	Vmware_2c:9b:a7	Broadcast	ARP	1	60 Who has 192.168.0.1
36	2019-10-25 10:02:13.050488	Cisco_f6:1d:ae	Vmware_2c:9b:a7	ARP	1	42 192.168.0.24
47	2019-10-25 10:02:19.284683	Vmware_2c:9b:a7	Broadcast	ARP	1	60 Who has 192.168.0.1
48	2019-10-25 10:02:19.284775	Cisco_f6:1d:ae	Vmware_2c:9b:a7	ARP	1	42 192.168.0.25
61	2019-10-25 10:02:25.779821	Vmware_2c:9b:a7	Broadcast	ARP	1	60 Who has 192.168.0.1
62	2019-10-25 10:02:25.779912	Cisco_f6:1d:ae	Vmware_2c:9b:a7	ARP	1	42 192.168.0.26
76	2019-10-25 10:02:31.978175	Vmware_2c:9b:a7	Broadcast	ARP	1	60 Who has 192.168.0.1
77	2019-10-25 10:02:31.978251	Cisco_f6:1d:ae	Vmware_2c:9b:a7	ARP	1	42 192.168.0.27
97	2019-10-25 10:02:38.666515	Vmware_2c:9b:a7	Broadcast	ARP	1	60 Who has 192.168.0.1
98	2019-10-25 10:02:38.666606	Cisco_f6:1d:ae	Vmware_2c:9b:a7	ARP	1	42 192.168.0.28
121	2019-10-25 10:02:47.384074	Vmware_2c:9b:a7	Broadcast	ARP	1	60 Who has 192.168.0.1
122	2019-10-25 10:02:47.384150	Cisco_f6:1d:ae	Vmware_2c:9b:a7	ARP	1	42 192.168.0.29
137	2019-10-25 10:02:53.539995	Vmware_2c:9b:a7	Broadcast	ARP	1	60 Who has 192.168.0.1
138	2019-10-25 10:02:53.540087	Cisco_f6:1d:ae	Vmware_2c:9b:a7	ARP	1	42 192.168.0.30

Belangrijkste punten:

1. De firewall ontvangt verschillende ARP-verzoeken om IP's binnen het 192.168.0.x/24-netwerk
2. De firewall antwoordt op alle (proxy-ARP) met zijn eigen MAC-adres

Aanbevolen acties

De acties die in deze paragraaf worden opgesomd, hebben tot doel de kwestie verder af te zwakken.

Actie 1. Controleer de NAT-configuratie.

Met betrekking tot de NAT configuratie, zijn er gevallen waar het **geen-volmacht-arp** sleutelwoord het vroegere gedrag kan verhinderen:

```
<#root>
```

```
firepower#
```

```
show run nat
```

```
nat (INSIDE,OUTSIDE) source static NET_1.1.1.0 NET_2.2.2.0 destination static NET_192.168.0.0 NET_4.4.4.4
```

```
no-proxy-arp
```

Actie 2. Schakel de proxy-arp functionaliteit op de firewall-interface uit.

Als het `no-proxy-arp` sleutelwoord het probleem niet oplost, probeer dan proxy ARP op de interface zelf uit te schakelen. In het geval van FTD moet u tijdens het schrijven FlexConfig gebruiken en de opdracht implementeren (de juiste interfacenaam opgeven).

```
sysopt noproxyarp INSIDE
```

Situatie 13. Identificeer SNMP-objectidentificatiecodes (OID's) die CPU-fouten veroorzaken

Deze case toont aan hoe bepaalde SNMP OID's voor geheugenpolling werden geïdentificeerd als de oorzaak van CPU-fouten (prestatiekwestie) op basis van de analyse van SNMP versie 3 (SNMPv3)-pakketvastlegging.

Probleem Beschrijving: Overschrijdingen op data interfaces nemen voortdurend toe. Verder onderzoek toonde aan dat er ook CPU-weblokken zijn (veroorzaakt door het SNMP-proces) die de grondoorzaak zijn van de interfaceoverschrijdingen.

De volgende stap in het probleemoplossingsproces was om de oorzaak van de CPU-fouten te identificeren die door het SNMP-proces werden veroorzaakt en in het bijzonder om de omvang van het probleem te beperken om de SNMP Object Identifiers (OID) te identificeren die, wanneer ze werden gepolijst, mogelijk konden resulteren in CPU-fouten.

Op dit moment biedt de FTD LINA engine geen 'show'-opdracht voor SNMP OID's die in real-time worden gepolled.

De lijst van SNMP OIDs voor opiniepeilingen kan uit het SNMP controle hulpmiddel worden teruggewonnen, echter, in dit geval, waren deze preventieve factoren:

- De FTD-beheerder had geen toegang tot de SNMP-bewakingstool
- SNMP versie 3 met verificatie en gegevenscodering voor privacy is geconfigureerd op FTD

Capture Analysis

Aangezien de FTD-beheerder de referenties had voor de verificatie en codering van SNMP versie 3, werd dit actieplan voorgesteld:

1. Neem SNMP-pakketopnamen
2. Sla de opnamen op en gebruik de voorkeuren voor het Wireshark SNMP-protocol om de SNMP versie 3-referenties te specificeren voor het decoderen van de SNMP versie 3-pakketten. De gedecrypteerde opnamen worden gebruikt voor de analyse en het ophalen van SNMP OID's

Configureer SNMP-pakketopnamen op de interface die wordt gebruikt in de hostconfiguratie van de SNMP-server:

```
<#root>
firepower#
show run snmp-server | include host
snmp-server host management 192.168.10.10 version 3 netmonv3

firepower#
show ip address management

System IP Address:
Interface          Name          IP address      Subnet mask      Method
Management0/0     management    192.168.5.254   255.255.255.0    CONFIG
Current IP Address:
Interface          Name          IP address      Subnet mask      Method
Management0/0     management    192.168.5.254   255.255.255.0    CONFIG

firepower#
capture capsnpmp interface management buffer 10000000 match udp host 192.168.10.10 host 192.168.5.254 eq

firepower#
show capture capsnpmp

capture capsnpmp type raw-data buffer 10000000 interface outside [Capturing -
9512
bytes]
match udp host 192.168.10.10 host 192.168.5.254 eq snmp
```

No.	Time	Protocol	Source	Source Port	Destination Port	Destination	Length	Info
1	0.000	SNMP	192.168.10.10	65484	161	192.168.5.254	100	getBulkRequest
2	0.000	SNMP	192.168.5.254	161	65484	192.168.10.10	167	report 1.3.6.1.6.3.15.1.1.4.0
3	0.176	SNMP	192.168.10.10	65484	161	192.168.5.254	197	encryptedPDU: privKey Unknown
4	0.176	SNMP	192.168.5.254	161	65484	192.168.10.10	192	report 1.3.6.1.6.3.15.1.1.2.0
5	0.325	SNMP	192.168.10.10	65484	161	192.168.5.254	199	encryptedPDU: privKey Unknown
6	0.326	SNMP	192.168.5.254	161	65484	192.168.10.10	678	encryptedPDU: privKey Unknown
7	0.490	SNMP	192.168.10.10	65484	161	192.168.5.254	205	encryptedPDU: privKey Unknown
8	0.490	SNMP	192.168.5.254	161	65484	192.168.10.10	560	encryptedPDU: privKey Unknown
9	0.675	SNMP	192.168.10.10	65484	161	192.168.5.254	205	encryptedPDU: privKey Unknown
10	0.767	SNMP	192.168.5.254	161	65484	192.168.10.10	610	encryptedPDU: privKey Unknown
11	0.945	SNMP	192.168.10.10	65484	161	192.168.5.254	205	encryptedPDU: privKey Unknown
12	0.946	SNMP	192.168.5.254	161	65484	192.168.10.10	584	encryptedPDU: privKey Unknown
13	1.133	SNMP	192.168.10.10	65484	161	192.168.5.254	205	encryptedPDU: privKey Unknown
14	1.134	SNMP	192.168.5.254	161	65484	192.168.10.10	588	encryptedPDU: privKey Unknown
15	1.317	SNMP	192.168.10.10	65484	161	192.168.5.254	205	encryptedPDU: privKey Unknown
16	1.318	SNMP	192.168.5.254	161	65484	192.168.10.10	513	encryptedPDU: privKey Unknown
17	17.595	SNMP	192.168.10.10	62008	161	192.168.5.254	100	getBulkRequest
18	17.595	SNMP	192.168.5.254	161	62008	192.168.10.10	167	report 1.3.6.1.6.3.15.1.1.4.0
19	17.749	SNMP	192.168.10.10	62008	161	192.168.5.254	197	encryptedPDU: privKey Unknown
20	17.749	SNMP	192.168.5.254	161	62008	192.168.10.10	192	report 1.3.6.1.6.3.15.1.1.2.0
21	17.898	SNMP	192.168.10.10	62008	161	192.168.5.254	199	encryptedPDU: privKey Unknown
22	17.899	SNMP	192.168.5.254	161	62008	192.168.10.10	678	encryptedPDU: privKey Unknown
23	18.094	SNMP	192.168.10.10	62008	161	192.168.5.254	205	encryptedPDU: privKey Unknown
24	18.094	SNMP	192.168.5.254	161	62008	192.168.10.10	560	encryptedPDU: privKey Unknown
25	18.290	SNMP	192.168.10.10	62008	161	192.168.5.254	205	encryptedPDU: privKey Unknown


```

<[Destination Host: 192.168.5.254]>
<[Source or Destination Host: 192.168.5.254]>
> User Datagram Protocol, Src Port: 65484, Dst Port: 161
v Simple Network Management Protocol
  msgVersion: snmpv3 (3)
  > msgGlobalData
  > msgAuthoritativeEngineID: 80000009fe1c6dad4930a00ef1fec2301621a4158bfc1f40...
  msgAuthoritativeEngineBoots: 0
  msgAuthoritativeEngineTime: 0
  msgUserName: netmonv3
  msgAuthenticationParameters: ff5176f5973c30b62ffc11b8
  msgPrivacyParameters: 000040e100003196
  v msgData: encryptedPDU (1)
    encryptedPDU: 879a16d23633400a0391c5280d226e0cec844d87101ba703...

```

Belangrijkste punten:

1. SNMP-bron- en doeladressen/-poorten.
2. Het SNMP-protocol PDU kan niet worden gedecodeerd omdat privKey niet bekend is bij Wireshark.
3. De waarde van de versleutelde PDU-primitief.

Aanbevolen acties

De acties die in deze paragraaf worden opgesomd, hebben tot doel de kwestie verder af te zwakken.

Actie 1. Decrypteer de SNMP-opnamen.

Sla de opnamen op en bewerk de voorkeuren voor het Wireshark SNMP-protocol om de SNMP versie 3-referenties te specificeren voor het decoderen van de pakketten.

```
<#root>
```

```
firepower#
```

```
copy /pcap capture: tftp:
```

```
Source capture name [capsnmp]?
```

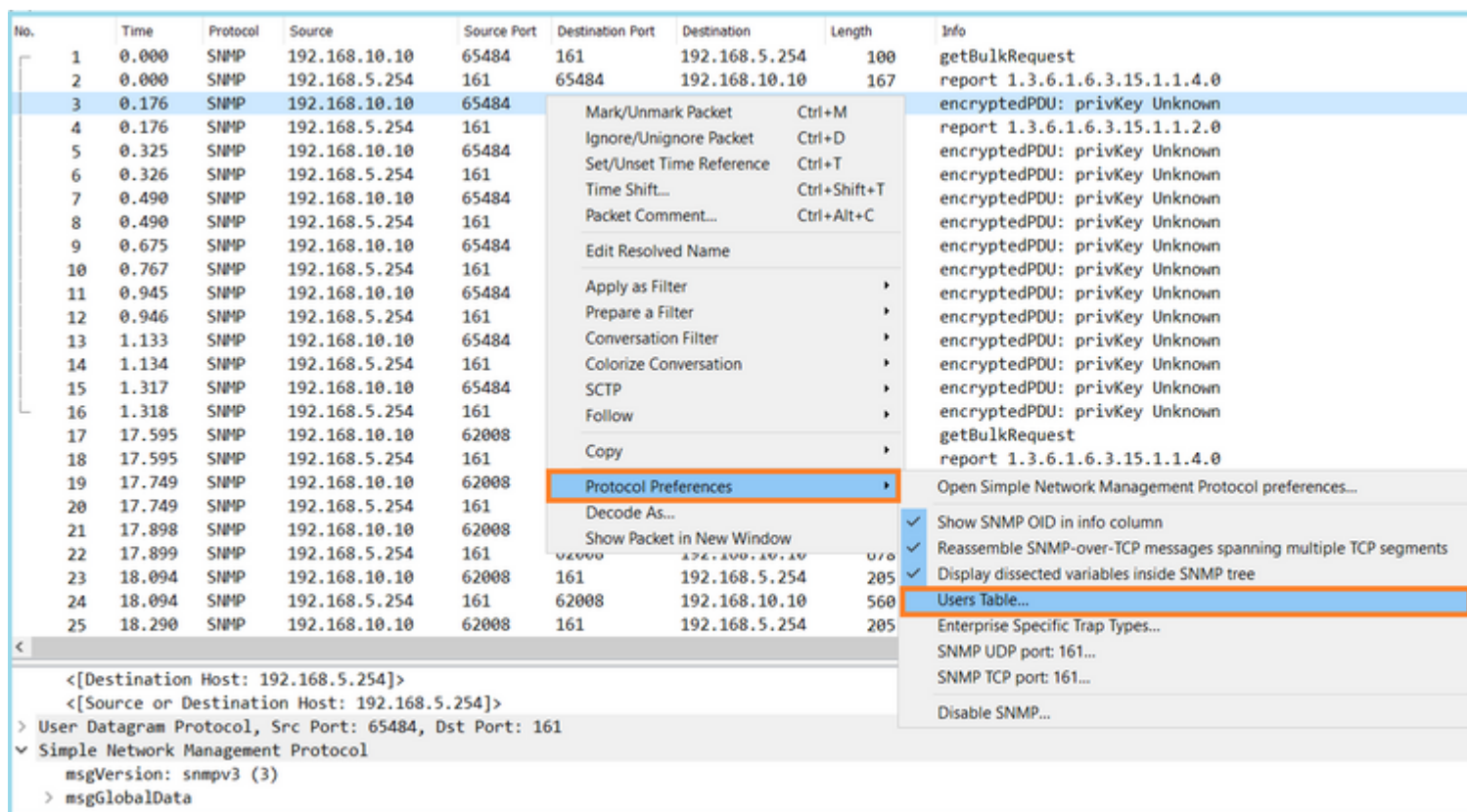
```
Address or name of remote host []? 192.168.10.253
```

Destination filename [capsnmp]? capsnmp.pcap

!!!!!!

64 packets copied in 0.40 secs

Open het opnamebestand op Wireshark, selecteer een SNMP-pakket en navigeer naar **Protocolvoorkeuren** > **Gebruikers-tabel**, zoals in de afbeelding:



In de SNMP-gebruikerstabel zijn de gebruikersnaam, het verificatiemodel, het verificatiewachtwoord, het privacyprotocol en het privacywachtwoord voor de SNMP versie 3 gespecificeerd (werkelijke referenties worden hieronder niet weergegeven):

SNMP Users

Engine ID	Username	Authentication model	Password	Privacy protocol	Privacy password
		MD5		DES	

<C:\Users\igasimov\AppData\Roaming\Wireshark\profiles\Profile1>

Zodra SNMP-gebruikers instellingen werden toegepast, toonde Wireshark gedecrypteerde SNMP PDU's:

No.	Time	Protocol	Source	Source Port	Destination Port	Destination	Length	Info
1	0.000	SNMP	192.168.10.10	65484	161	192.168.5.254	100	1 getBulkRequest
2	0.000	SNMP	192.168.5.254	161	65484	192.168.10.10	167	report 1.3.6.1.6.3.15.1.1.4.0
3	0.176	SNMP	192.168.10.10	65484	161	192.168.5.254	197	1 getBulkRequest 1.3.6.1.4.1.9.9.221.1
4	0.176	SNMP	192.168.5.254	161	65484	192.168.10.10	192	report 1.3.6.1.6.3.15.1.1.2.0
5	0.325	SNMP	192.168.10.10	65484	161	192.168.5.254	199	1 getBulkRequest 1.3.6.1.4.1.9.9.221.1
6	0.326	SNMP	192.168.5.254	161	65484	192.168.10.10	678	2 get-response 1.3.6.1.4.1.9.9.221.1.1.1.2.1.1 1.3.6.1.4.1.9.9.221.1.1.1.3.1.8
7	0.490	SNMP	192.168.10.10	65484	161	192.168.5.254	205	1 getBulkRequest 1.3.6.1.4.1.9.9.221.1.1.1.1.8.1.8
8	0.490	SNMP	192.168.5.254	161	65484	192.168.10.10	560	2 get-response 1.3.6.1.4.1.9.9.221.1.1.1.5.1.1 1.3.6.1.4.1.9.9.221.1.1.1.6.1.8
9	0.675	SNMP	192.168.10.10	65484	161	192.168.5.254	205	1 getBulkRequest 1.3.6.1.4.1.9.9.221.1.1.1.1.6.1.8
10	0.767	SNMP	192.168.5.254	161	65484	192.168.10.10	610	2 get-response 1.3.6.1.4.1.9.9.221.1.1.1.7.1.1 1.3.6.1.4.1.9.9.221.1.1.1.8.1.8
11	0.945	SNMP	192.168.10.10	65484	161	192.168.5.254	205	1 getBulkRequest 1.3.6.1.4.1.9.9.221.1.1.1.1.8.1.8
12	0.946	SNMP	192.168.5.254	161	65484	192.168.10.10	584	2 get-response 1.3.6.1.4.1.9.9.221.1.1.1.17.1.1 1.3.6.1.4.1.9.9.221.1.1.1.19.1.1 1.3.6.1.4.1.9.9.221.1.1.1.20.1.8
13	1.133	SNMP	192.168.10.10	65484	161	192.168.5.254	205	1 getBulkRequest 1.3.6.1.4.1.9.9.221.1.1.1.1.18.1.8
14	1.134	SNMP	192.168.5.254	161	65484	192.168.10.10	588	2 get-response 1.3.6.1.4.1.9.9.221.1.1.1.1.19.1.1 1.3.6.1.4.1.9.9.221.1.1.1.20.1.8
15	1.317	SNMP	192.168.10.10	65484	161	192.168.5.254	205	1 getBulkRequest 1.3.6.1.4.1.9.9.221.1.1.1.1.20.1.8
16	1.318	SNMP	192.168.5.254	161	65484	192.168.10.10	513	2 get-response 1.3.6.1.4.1.9.9.392.1.1.1.0 1.3.6.1.4.1.9.9.221.1.1.1.1.4.0
17	17.595	SNMP	192.168.10.10	62008	161	192.168.5.254	100	getBulkRequest
18	17.595	SNMP	192.168.5.254	161	62008	192.168.10.10	167	report 1.3.6.1.6.3.15.1.1.4.0
19	17.749	SNMP	192.168.10.10	62008	161	192.168.5.254	197	1 getBulkRequest 1.3.6.1.4.1.9.9.221.1
20	17.749	SNMP	192.168.5.254	161	62008	192.168.10.10	192	report 1.3.6.1.6.3.15.1.1.2.0
21	17.898	SNMP	192.168.10.10	62008	161	192.168.5.254	199	1 getBulkRequest 1.3.6.1.4.1.9.9.221.1
22	17.899	SNMP	192.168.5.254	161	62008	192.168.10.10	678	2 get-response 1.3.6.1.4.1.9.9.221.1.1.1.2.1.1 1.3.6.1.4.1.9.9.221.1.1.1.3.1.8
23	18.094	SNMP	192.168.10.10	62008	161	192.168.5.254	205	1 getBulkRequest 1.3.6.1.4.1.9.9.221.1.1.1.1.3.1.8
24	18.094	SNMP	192.168.5.254	161	62008	192.168.10.10	560	2 get-response 1.3.6.1.4.1.9.9.221.1.1.1.5.1.1 1.3.6.1.4.1.9.9.221.1.1.1.6.1.8
25	18.290	SNMP	192.168.10.10	62008	161	192.168.5.254	205	1 getBulkRequest 1.3.6.1.4.1.9.9.221.1.1.1.1.6.1.8

```

msgData: encryptedPDU (1)
  encryptedPDU: 879a16d23633400a0391c5280d226e0cec844d87101ba703...
    Decrypted ScopedPDU: 303b04198000009fec1c6dad4930a00ef1fec2301621a415...
      contextEngineID: 80000009fec1c6dad4930a00ef1fec2301621a4158bfc1f40...
      contextName:
      data: getBulkRequest (5)
        getBulkRequest
          request-id: 5620
          non-repeaters: 0
          max-repetitions: 16
          variable-bindings: 1 item
            1.3.6.1.4.1.9.9.221.1: Value (Null)
              Object Name: 1.3.6.1.4.1.9.9.221.1 (iso.3.6.1.4.1.9.9.221.1)
              Value (Null)
  
```

Belangrijkste punten:

1. De SNMP-monitoringtools gebruiken SNMP getBulkVerzoek om te vragen en te lopen over de ouder OID 1.3.6.1.4.1.9.9.221.1 en verwante OIDs.
2. Het FTD reageerde op elke getBulkrequest met get-response die OID's bevat die gerelateerd zijn aan 1.3.6.1.4.1.9.9.221.1.

Actie 2. Identificeer de SNMP-OID's.

[SNMP Object Navigator](#) heeft aangetoond dat OID 1.3.6.1.4.1.9.9.221.1 behoort tot de Management Information Base (MIB) met de naam **CISCO-ENHANCED-MEMPOOL-MIB**, zoals in de afbeelding:

The screenshot shows the 'SNMP Object Navigator' interface. At the top, there are navigation tabs: 'HOME', 'SUPPORT', and 'TOOLS & RESOURCES', with 'SNMP Object Navigator' selected. Below the navigation is a search bar with the text 'Translate | Browse The Object Tree'. The main input field contains the OID '1.3.6.1.4.1.9.9.221.1' and a 'Translate' button. To the right, there are examples: 'examples - OID: 1.3.6.1.4.1.9.9.27 Object Name: ifIndex'. Below the input field, the 'Object Information' section is displayed, showing a table with the following data:

Specific Object Information	
Object	cempMIBObjects
OID	1.3.6.1.4.1.9.9.221.1
MIB	CISCO-ENHANCED-MEMPOOL-MIB ; - View Supporting Images

Below the table, the 'OID Tree' section shows the hierarchy: '. iso (1). org (3). dod (6). internet (1). private (4). enterprises (1). cisco (9)'. Under 'cisco (9)', there are two sub-entries: '- -- ciscoMgmt (9)' and '+ -- ciscoTcpMIB (6)'. The 'ciscoMgmt (9)' entry is expanded, showing the 'ciscoTcpMIB (6)' entry.

U kunt de OID's als volgt in een door mensen leesbaar formaat weergeven in Wireshark:

1. Download de MIB **CISCO-ENHANCED-MEMPOOL-MIB** en de afhankelijkheden ervan, zoals in de afbeelding:

SNMP Object Navigator

[HOME](#)

[SUPPORT](#)

[TOOLS & RESOURCES](#)

SNMP Object Navigator

TRANSLATE/BROWSE

SEARCH

DOWNLOAD MIBS

MIB SUPPORT - SW

View MIB dependencies and download MIB or view MIB contents

Step 1. Select a MIB name by typing or scrolling and then select a function in step 2 and click Submit

CISCO-ENHANCED-MEMPOOL-MIB

List matching MIBs

A100-R1-MIB
ACCOUNTING-CONTROL-MIB
ACTONA-ACTASTOR-MIB
ADMIN-AUTH-STATS-MIB
ADSL-DMT-LINE-MIB
ADSL-LINE-MIB
ADSL-TC-MIB
ADSL2-LINE-MIB

Step 2: Select a function:

- View MIB dependencies and download MIB
- View MIB contents

Submit

SNMP Object Navigator

[HOME](#)[SUPPORT](#)[TOOLS & RESOURCES](#)**SNMP Object Navigator**[TRANSLATE/BROWSE](#)[SEARCH](#)[DOWNLOAD MIBS](#)[MIB SUPPORT - SW](#)**CISCO-ENHANCED-MEMPOOL-MIB**

View compiling dependencies for other MIBs by [clearing](#) the page and selecting another MIB.

Compile the MIB

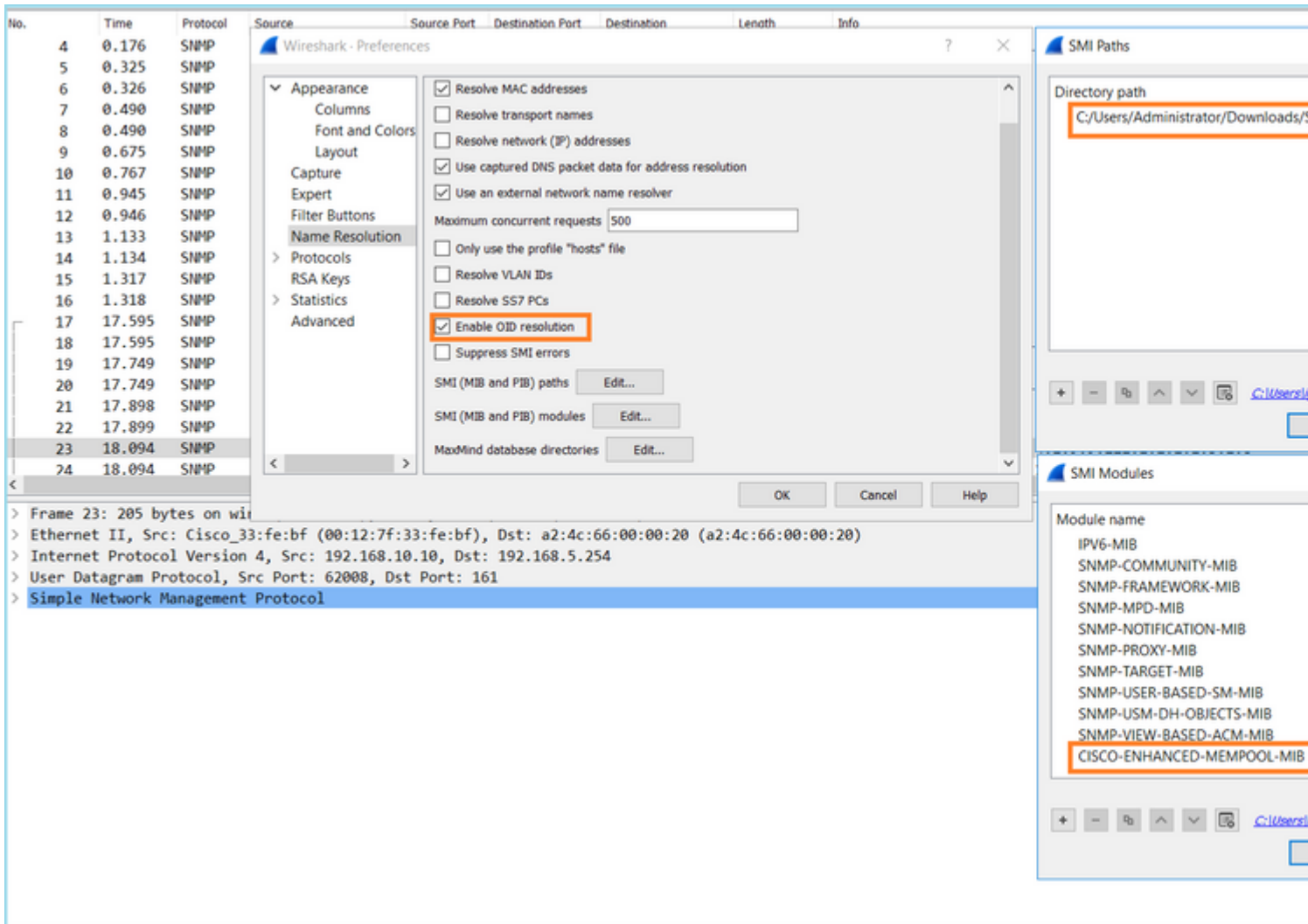
Before you can compile CISCO-ENHANCED-MEMPOOL-MIB, you need to compile the MIBs listed below in the order listed.

Download all of these MIBs (Warning: does not include non-Cisco MIBs) or view details about each MIB below.

If you are using Internet Explorer click [here](#).

MIB Name	Version 1	Version 2	Dependencies
1. SNMPv2-SMI	Download	Download	View Dependencies
2. SNMPv2-TC	Download	Download	View Dependencies
3. SNMPv2-CONF	Not Required	Download	View Dependencies
4. SNMP-FRAMEWORK-MIB	Download	Download	View Dependencies
5. CISCO-SMI	Download	Download	View Dependencies
6. ENTITY-MIB	Download	Download	View Dependencies
7. HCNUM-TC	Download	Download	View Dependencies
8. RFC1155-SMI	Non-Cisco MIB	Non-Cisco MIB	-
9. RFC-1212	Non-Cisco MIB	Non-Cisco MIB	-
10. RFC-1215	Non-Cisco MIB	Non-Cisco MIB	-
11. SNMPv2-TC-v1	Non-Cisco MIB	Non-Cisco MIB	-
12. CISCO-ENHANCED-MEMPOOL-MIB	Download	Download	

2. In Wireshark in **Edit > Preferences > Name Resolution** venster wordt de optie **OID-resolutie inschakelen** ingeschakeld. In het venster **SMI (MIB- en PIB-paden)** specificeert u de map met de gedownload MIB's en in **SMI (MIB- en PIB-modules)**. Cisco-ENHANCED-MEMPOOL-MIB wordt automatisch toegevoegd aan de lijst met modules:



3. Zodra Wireshark opnieuw is gestart, wordt de OID-resolutie geactiveerd:

No.	Time	Protocol	Source	Source Port	Destination Port	Destination	Length	Info
1	0.000	SNMP	192.168.10.10	65484	161	192.168.5.254	100	getBulkRequest
2	0.000	SNMP	192.168.5.254	161	65484	192.168.10.10	167	report SNMP-USER-BASED-SM-MIB::usmStatsUnknownEngineID
3	0.176	SNMP	192.168.10.10	65484	161	192.168.5.254	197	getBulkRequest CISCO-ENHANCED-MEMPOOL-MIB::cempMIBObj
4	0.176	SNMP	192.168.5.254	161	65484	192.168.10.10	192	report SNMP-USER-BASED-SM-MIB::usmStatsNotInTimeWindow
5	0.325	SNMP	192.168.10.10	65484	161	192.168.5.254	199	getBulkRequest CISCO-ENHANCED-MEMPOOL-MIB::cempMIBObj
6	0.326	SNMP	192.168.5.254	161	65484	192.168.10.10	678	get-response CISCO-ENHANCED-MEMPOOL-MIB::cempMemPoolTyp
7	0.490	SNMP	192.168.10.10	65484	161	192.168.5.254	205	getBulkRequest CISCO-ENHANCED-MEMPOOL-MIB::cempMemPoolAl
8	0.490	SNMP	192.168.5.254	161	65484	192.168.10.10	560	get-response CISCO-ENHANCED-MEMPOOL-MIB::cempMemPoolAl
9	0.675	SNMP	192.168.10.10	65484	161	192.168.5.254	205	getBulkRequest CISCO-ENHANCED-MEMPOOL-MIB::cempMemPool
10	0.767	SNMP	192.168.5.254	161	65484	192.168.10.10	610	get-response CISCO-ENHANCED-MEMPOOL-MIB::cempMemPoolUs
11	0.945	SNMP	192.168.10.10	65484	161	192.168.5.254	205	getBulkRequest CISCO-ENHANCED-MEMPOOL-MIB::cempMemPool
12	0.946	SNMP	192.168.5.254	161	65484	192.168.10.10	584	get-response CISCO-ENHANCED-MEMPOOL-MIB::cempMemPoolUs
13	1.133	SNMP	192.168.10.10	65484	161	192.168.5.254	205	getBulkRequest CISCO-ENHANCED-MEMPOOL-MIB::cempMemPool
14	1.134	SNMP	192.168.5.254	161	65484	192.168.10.10	588	get-response CISCO-ENHANCED-MEMPOOL-MIB::cempMemPoolE

```

v CISCO-ENHANCED-MEMPOOL-MIB::cempMemPoolName.1.1 (1.3.6.1.4.1.9.9.221.1.1.1.3.1.1): System memory
  Object Name: 1.3.6.1.4.1.9.9.221.1.1.1.3.1.1 (CISCO-ENHANCED-MEMPOOL-MIB::cempMemPoolName.1.1)
  CISCO-ENHANCED-MEMPOOL-MIB::cempMemPoolName: System memory
v CISCO-ENHANCED-MEMPOOL-MIB::cempMemPoolName.1.2 (1.3.6.1.4.1.9.9.221.1.1.1.3.1.2): System memory
  Object Name: 1.3.6.1.4.1.9.9.221.1.1.1.3.1.2 (CISCO-ENHANCED-MEMPOOL-MIB::cempMemPoolName.1.2)
  CISCO-ENHANCED-MEMPOOL-MIB::cempMemPoolName: System memory
v CISCO-ENHANCED-MEMPOOL-MIB::cempMemPoolName.1.3 (1.3.6.1.4.1.9.9.221.1.1.1.3.1.3): MEMPOOL_MSGLYR
  Object Name: 1.3.6.1.4.1.9.9.221.1.1.1.3.1.3 (CISCO-ENHANCED-MEMPOOL-MIB::cempMemPoolName.1.3)
  CISCO-ENHANCED-MEMPOOL-MIB::cempMemPoolName: MEMPOOL_MSGLYR
v CISCO-ENHANCED-MEMPOOL-MIB::cempMemPoolName.1.4 (1.3.6.1.4.1.9.9.221.1.1.1.3.1.4): MEMPOOL_HEAPCACHE_1
  Object Name: 1.3.6.1.4.1.9.9.221.1.1.1.3.1.4 (CISCO-ENHANCED-MEMPOOL-MIB::cempMemPoolName.1.4)
  CISCO-ENHANCED-MEMPOOL-MIB::cempMemPoolName: MEMPOOL_HEAPCACHE_1
v CISCO-ENHANCED-MEMPOOL-MIB::cempMemPoolName.1.5 (1.3.6.1.4.1.9.9.221.1.1.1.3.1.5): MEMPOOL_HEAPCACHE_0
  Object Name: 1.3.6.1.4.1.9.9.221.1.1.1.3.1.5 (CISCO-ENHANCED-MEMPOOL-MIB::cempMemPoolName.1.5)
  CISCO-ENHANCED-MEMPOOL-MIB::cempMemPoolName: MEMPOOL_HEAPCACHE_0
v CISCO-ENHANCED-MEMPOOL-MIB::cempMemPoolName.1.6 (1.3.6.1.4.1.9.9.221.1.1.1.3.1.6): MEMPOOL_DMA_ALT1
  Object Name: 1.3.6.1.4.1.9.9.221.1.1.1.3.1.6 (CISCO-ENHANCED-MEMPOOL-MIB::cempMemPoolName.1.6)
  CISCO-ENHANCED-MEMPOOL-MIB::cempMemPoolName: MEMPOOL_DMA_ALT1
v CISCO-ENHANCED-MEMPOOL-MIB::cempMemPoolName.1.7 (1.3.6.1.4.1.9.9.221.1.1.1.3.1.7): MEMPOOL_DMA
  Object Name: 1.3.6.1.4.1.9.9.221.1.1.1.3.1.7 (CISCO-ENHANCED-MEMPOOL-MIB::cempMemPoolName.1.7)
  CISCO-ENHANCED-MEMPOOL-MIB::cempMemPoolName: MEMPOOL_DMA
v CISCO-ENHANCED-MEMPOOL-MIB::cempMemPoolName.1.8 (1.3.6.1.4.1.9.9.221.1.1.1.3.1.8): MEMPOOL_GLOBAL_SHARED
  Object Name: 1.3.6.1.4.1.9.9.221.1.1.1.3.1.8 (CISCO-ENHANCED-MEMPOOL-MIB::cempMemPoolName.1.8)
  CISCO-ENHANCED-MEMPOOL-MIB::cempMemPoolName: MEMPOOL_GLOBAL_SHARED

```

Op basis van de gedecrypteerde uitvoer van het opnamebestand was de SNMP-bewakingstool periodiek (10 seconden interval) opiniegegevens over het gebruik van geheugenpools op de FTD. Zoals uitgelegd in het TechNote-artikel [ASA SNMP Polling for Memory-related Statistics](#), resulteert het pollen van het gebruik van Global Shared Pool (GSP) met SNMP in een hoog CPU-gebruik. In dit geval van de Captures, was het duidelijk dat de Global Shared Pool gebruik werd periodiek ondervraagd als deel van SNMP getBulkRequest primitief.

Om de CPU-fouten die door het SNMP-proces worden veroorzaakt tot een minimum te beperken, werd aanbevolen de in het artikel vermelde mitigatiestappen voor de CPU-fouten voor SNMP te volgen en niet de OID's met betrekking tot het SAP te hoeven opvragen. Zonder de SNMP-enquête voor de OID's die betrekking hebben op het SAP, werden geen CPU-varkens als gevolg van het SNMP-proces waargenomen en nam het percentage overschrijdingen aanzienlijk af.

Gerelateerde informatie

- [Configuratiehandleidingen voor Cisco Firepower Management Center](#)
- [Inzicht in acties door beleidsregels inzake toegangscontrole van Firepower Threat Defense](#)
- [Werken met FirePOWER Threat Defence Captures en Packet Tracer](#)
- [Leer Wireshark](#)

Over deze vertaling

Cisco heeft dit document vertaald via een combinatie van machine- en menselijke technologie om onze gebruikers wereldwijd ondersteuningscontent te bieden in hun eigen taal. Houd er rekening mee dat zelfs de beste machinevertaling niet net zo nauwkeurig is als die van een professionele vertaler. Cisco Systems, Inc. is niet aansprakelijk voor de nauwkeurigheid van deze vertalingen en raadt aan altijd het oorspronkelijke Engelstalige document ([link](#)) te raadplegen.