

排除Firepower威脅防禦(FTD)群集故障

目錄

[簡介](#)

[必要條件](#)

[需求](#)

[採用元件](#)

[背景資訊](#)

[設定](#)

[集群基礎知識](#)

[NGFW架構](#)

[群集捕獲](#)

[集群控制鏈路\(CCL\)消息](#)

[集群控制點\(CCP\)消息](#)

[叢集健康狀況檢查\(HC\)機制](#)

[集群HC故障場景](#)

[建立群集資料平面連線](#)

[疑難排解](#)

[群集故障排除簡介](#)

[群集資料平面問題](#)

[NAT/PAT常見問題](#)

[片段處理](#)

[ACI問題](#)

[群集控制平面問題](#)

[裝置無法加入群集](#)

[CCL上的MTU大小](#)

[集群裝置之間的介面不匹配](#)

[資料/埠通道介面問題](#)

[由於在CCL上的可達性問題導致大腦分裂](#)

[由於暫停的資料埠通道介面而禁用群集](#)

[群集穩定性問題](#)

[FXOS回溯](#)

[磁碟已滿](#)

[溢位保護](#)

[簡化模式](#)

[相關資訊](#)

簡介

本檔案將說明Firepower下一代防火牆(NGFW)上群集設定的疑難排解。

必要條件

需求

思科建議您瞭解以下主題（如需連結，請參閱相關資訊一節）：

- Firepower平台架構
- Firepower群集配置和操作
- 熟悉FTD和Firepower可擴展作業系統(FXOS)CLI
- NGFW/資料平面日誌
- NGFW/資料平面Packet Tracer
- FXOS/資料平面捕獲

採用元件

- 硬體：Firepower 4125
- SW:6.7.0（內部版本65）— 資料平面9.15(1)

本文中的資訊是根據特定實驗室環境內的裝置所建立。文中使用到的所有裝置皆從已清除（預設）的組態來啟動。如果您的網路運作中，請確保您瞭解任何指令可能造成的影響。

背景資訊

本文檔中介紹的大多數專案也完全適用於自適應安全裝置(ASA)群集故障排除。

設定

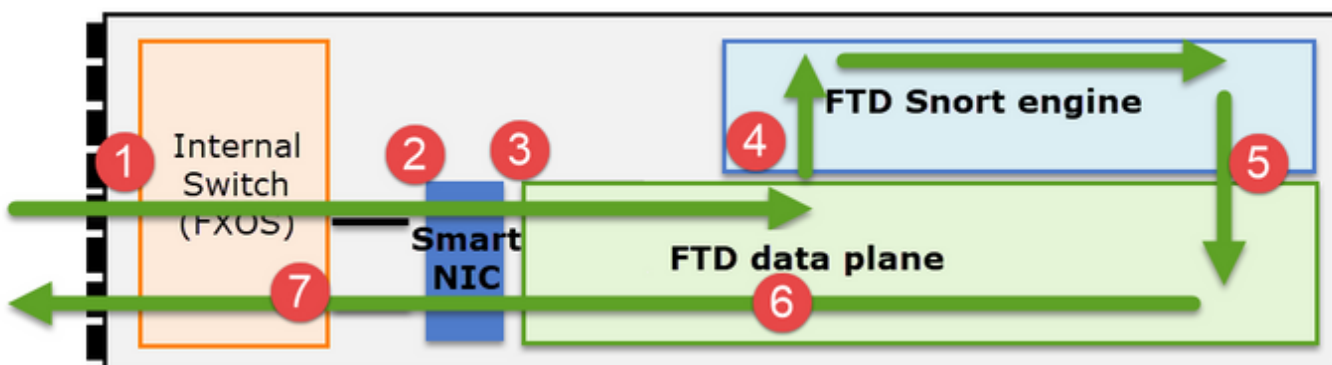
FMC和FXOS配置指南中介紹了集群部署的配置部分：

- [用於Firepower威脅防禦的群集](#)
- [部署用於Firepower威脅防禦的群集，以實現可擴充性和高可用性](#)

集群基礎知識

NGFW架構

瞭解Firepower 41xx或93xx系列如何處理傳輸資料包非常重要：



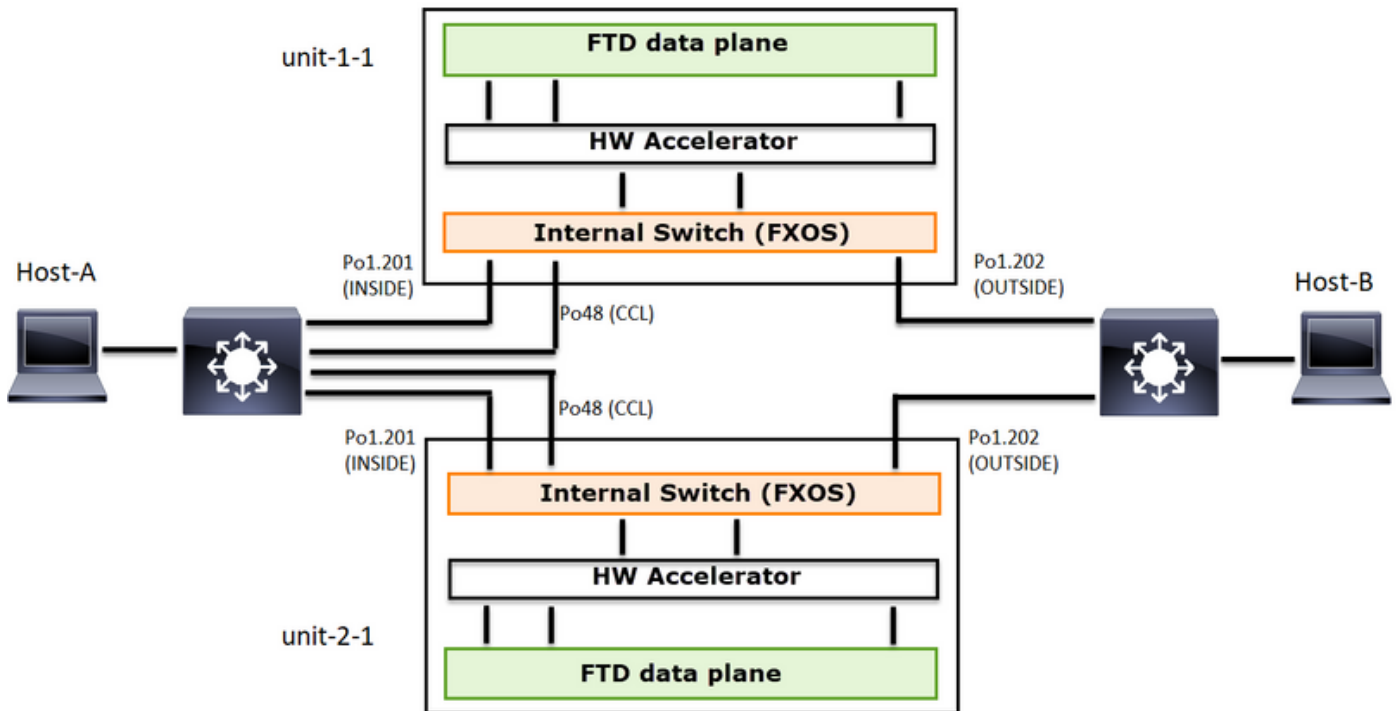
1. 封包進入輸入介面，並由機箱內部交換器處理。
2. 封包會通過智慧網絡卡。如果流量被分流（硬體加速），則資料包將僅由智慧NIC處理，然後被傳送回網路。
3. 如果封包沒有解除安裝，則會進入主要執行L3/L4檢查的FTD資料平面。
4. 如果原則需要該封包，則Snort引擎會對其進行檢查（主要是L7檢查）。
5. Snort引擎傳回封包的判定結果（例如允許或封鎖）。
6. 資料平面根據Snort的判定結果丟棄或轉發資料包。
7. 封包透過內部機箱交換器離開機箱。

群集捕獲

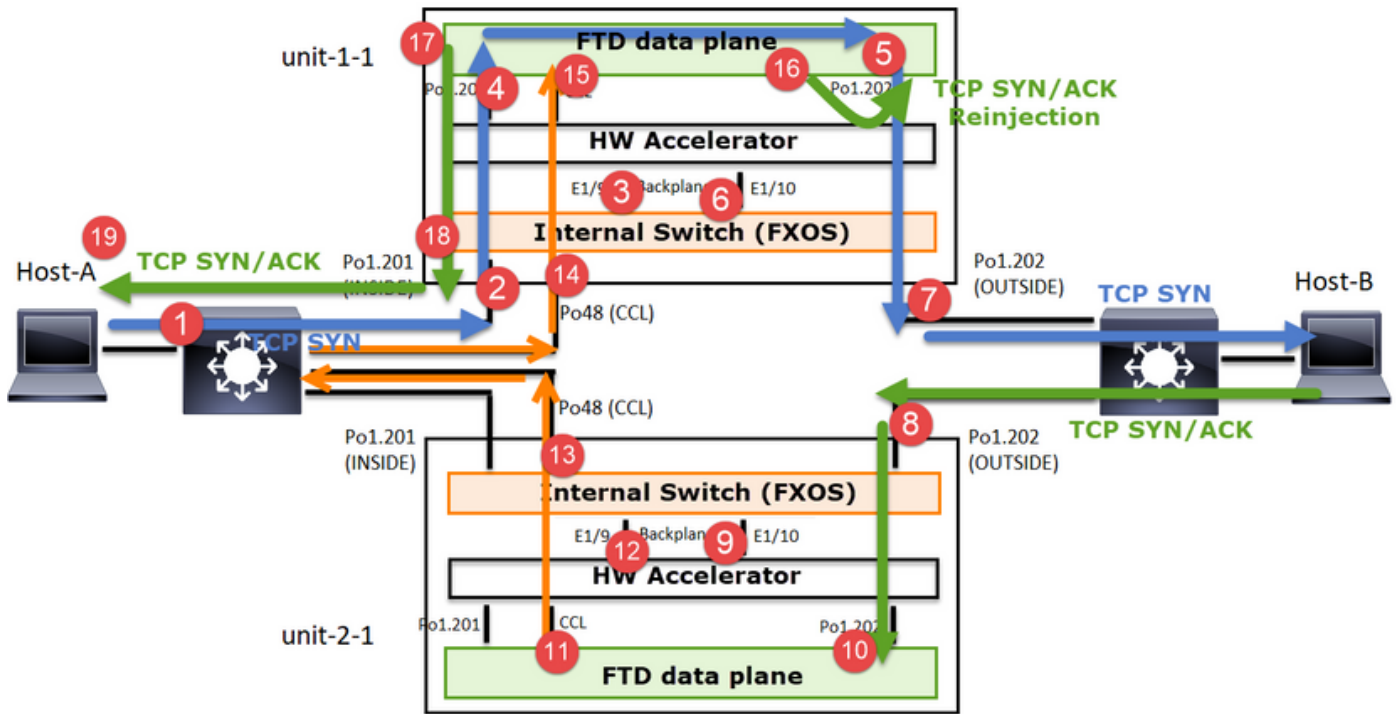
Firepower裝置提供多個捕獲點，用於檢視傳輸流。在排除故障和啟用群集捕獲時，主要挑戰如下：

- 捕獲數量隨著集群中裝置數量的增加而增加。
- 您需要瞭解群集處理特定流量的方式，才能跟蹤通過群集的資料包。

此圖顯示2單元群集（例如FP941xx/FP9300）：



建立非對稱TCP連線時，TCP SYN、SYN/ACK交換如下所示：



轉發流量

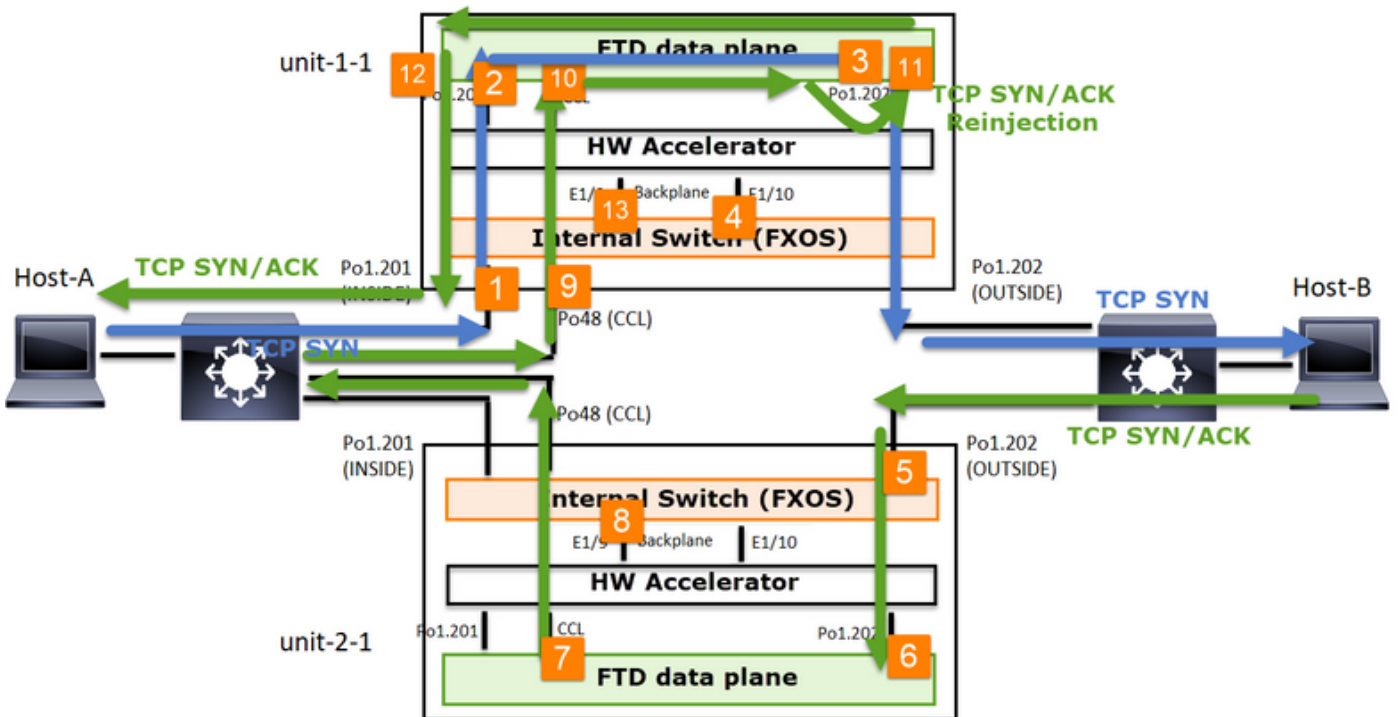
1. TCP SYN從主機A傳送到主機B。
2. TCP SYN到達機箱 (Po1的成員之一)。
3. TCP SYN透過其中一個機箱背板介面 (例如E1/9、 E1/10等) 傳送到資料平面。
4. TCP SYN到達資料平面輸入介面(Po1.201/INSIDE)。 在本示例中， unit1-1取得流的所有權，執行初始序列號(ISN)隨機化，並編碼序列號中的所有權(cookie)資訊。
5. TCP SYN從Po1.202/OUTSIDE (資料平面輸出介面) 發出。
6. TCP SYN到達其中一個機箱背板介面 (例如E1/9、 E1/10等)。
7. TCP SYN從機箱物理介面 (Po1的成員之一) 傳送到主機B。

返回流量

8. TCP SYN/ACK從主機B傳送並到達裝置2-1 (Po1的成員之一)。
9. TCP SYN/ACK透過其中一個機箱背板介面 (例如E1/9、 E1/10等) 傳送到資料平面。
10. TCP SYN/ACK到達資料平面輸入介面(Po1.202/OUTSIDE)。
11. TCP SYN/ACK從集群控制鏈路(CCL)傳送到裝置1-1。預設情況下， ISN處於啟用狀態。因此，轉發器可以在沒有指揮交換機參與的情況下找到TCP SYN+ACK的所有者資訊。對於其他資料包或當ISN被禁用時，會查詢導向器。
12. TCP SYN/ACK到達其中一個機箱背板介面 (例如E1/9、 E1/10等)。
13. TCP SYN/ACK從機箱物理介面 (Po48的成員之一) 傳送到裝置1-1。
14. TCP SYN/ACK到達裝置1-1 (Po48的成員之一)。
15. TCP SYN/ACK通過其中一個機箱背板介面轉發到資料平面CCL埠通道介面 (nameif集群)。
16. 資料平面將TCP SYN/ACK資料包重新傳送到資料平面介面Po1.202/OUTSIDE。
17. TCP SYN/ACK從Po1.201/INSIDE (資料平面輸出介面) 傳送到HOST-A。
18. TCP SYN/ACK會穿過其中一個機箱背板介面 (例如E1/9、 E1/10等)，並離開Po1的一個成員。
19. TCP SYN/ACK到達主機A。

有關此方案的更多詳細資訊，請參閱集群連線建立案例研究中的相關部分。

基於此資料包交換，所有可能的群集捕獲點包括：



對於上的轉發流量（例如TCP SYN）捕獲：

1. 機箱物理介面（例如Po1成員）。此捕獲是從機箱管理器(CM)UI或CM CLI配置的。
2. 資料平面輸入介面（例如Po1.201 INSIDE）。
3. 資料平面輸出介面（例如Po1.202 OUTSIDE）。
4. 機箱背板介面。FP4100上有2個背板介面。FP9300上共有6個（每個模組2個）。由於您不知道資料包到達哪個介面，因此必須在所有介面上啟用捕獲。


對於返回流量（例如TCP SYN/ACK），擷取：

5. 機箱物理介面（例如Po1成員）。此捕獲是從機箱管理器(CM)UI或CM CLI配置的。
6. 資料平面輸入介面（例如Po1.202 OUTSIDE）。
7. 由於資料包被重定向，因此下一個捕獲點是資料平面CCL。
8. 機箱背板介面。同樣，您必須在兩個介面上啟用捕獲。
9. Unit-1-1機箱CCL成員介面。
10. 資料平面CCL介面(nameif cluster)。
11. 輸入介面(Po1.202 OUTSIDE)。這是從CCL重新注入到資料平面的資料包。
12. 資料平面輸出介面（例如Po1.201 INSIDE）。
13. 機箱背板介面。

如何啟用群集捕獲

FXOS擷取

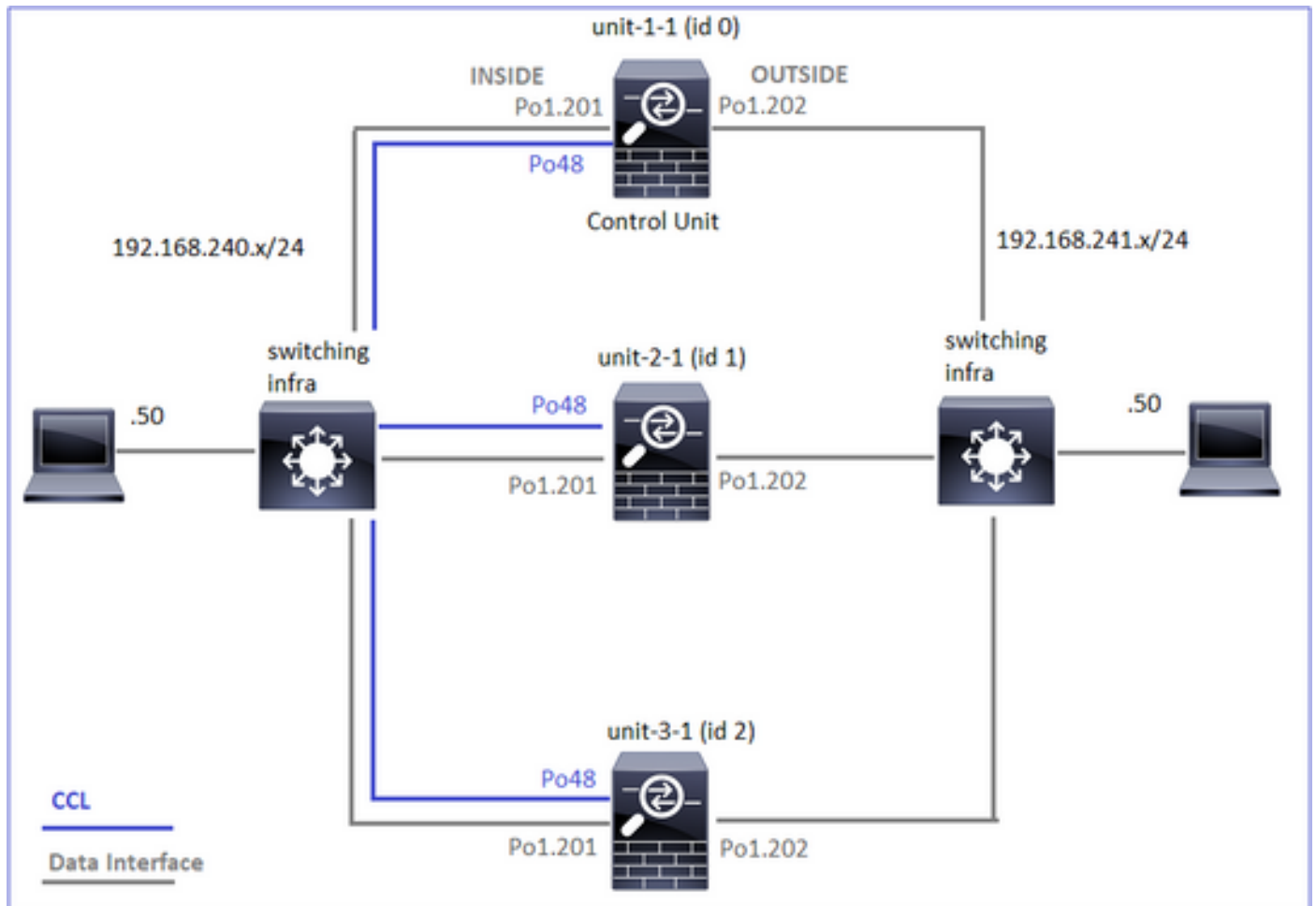
FXOS配置指南中介紹了該過程：[封包捕獲](#)

 附註：FXOS擷取只能從內部交換器的角度朝輸入方向進行。

資料平面捕獲

建議在所有群整合員上啟用捕獲的方法是使用cluster exec命令。

考慮一個包含3個單元的群集：



要驗證所有集群單元中是否存在活動捕獲，請使用以下命令：

```
<#root>
```

```
firepower#
```

```
cluster exec show capture
```

```
unit-1-1(LOCAL):*****
```

```
unit-2-1:*****
```

```
unit-3-1:*****
```

```
firepower#
```

要在Po1.201(INSIDE)上的所有裝置上啟用資料平面捕獲，請執行以下操作：

```
<#root>
firepower#
cluster exec capture CAPI interface INSIDE
```

強烈建議指定擷取過濾器，並在預期會有大量流量時，增加擷取緩衝區：

```
<#root>
firepower#
cluster exec capture CAPI buffer 33554432 interface INSIDE match tcp host 192.168.240.50 host 192.168.24
```

驗證

```
<#root>
firepower#
cluster exec show capture

unit-1-1(LOCAL):*****
capture CAPI type raw-data buffer 33554432 interface INSIDE [Capturing - 5140 bytes]
  match tcp host 192.168.240.50 host 192.168.241.50 eq www

unit-2-1:*****
capture CAPI type raw-data buffer 33554432 interface INSIDE [Capturing - 260 bytes]
  match tcp host 192.168.240.50 host 192.168.241.50 eq www

unit-3-1:*****
capture CAPI type raw-data buffer 33554432 interface INSIDE [Capturing - 0 bytes]
  match tcp host 192.168.240.50 host 192.168.241.50 eq www
```

要檢視所有捕獲的內容（此輸出可能很長）：

```
<#root>
firepower#
terminal pager 24

firepower#
cluster exec show capture CAPI
```

```
unit-1-1(LOCAL):*****
```

```
21 packets captured
```

```
1: 11:33:09.879226 802.1Q vlan#201 PO 192.168.240.50.45456 > 192.168.241.50.80: S 2225395909:2225395909
2: 11:33:09.880401 802.1Q vlan#201 PO 192.168.241.50.80 > 192.168.240.50.45456: S 719653963:719653963(0
3: 11:33:09.880691 802.1Q vlan#201 PO 192.168.240.50.45456 > 192.168.241.50.80: . ack 719653964 win 229
4: 11:33:09.880783 802.1Q vlan#201 PO 192.168.240.50.45456 > 192.168.241.50.80: P 2225395910:2225396054
```

```
unit-2-1:*****
```

```
0 packet captured
```

```
0 packet shown
```

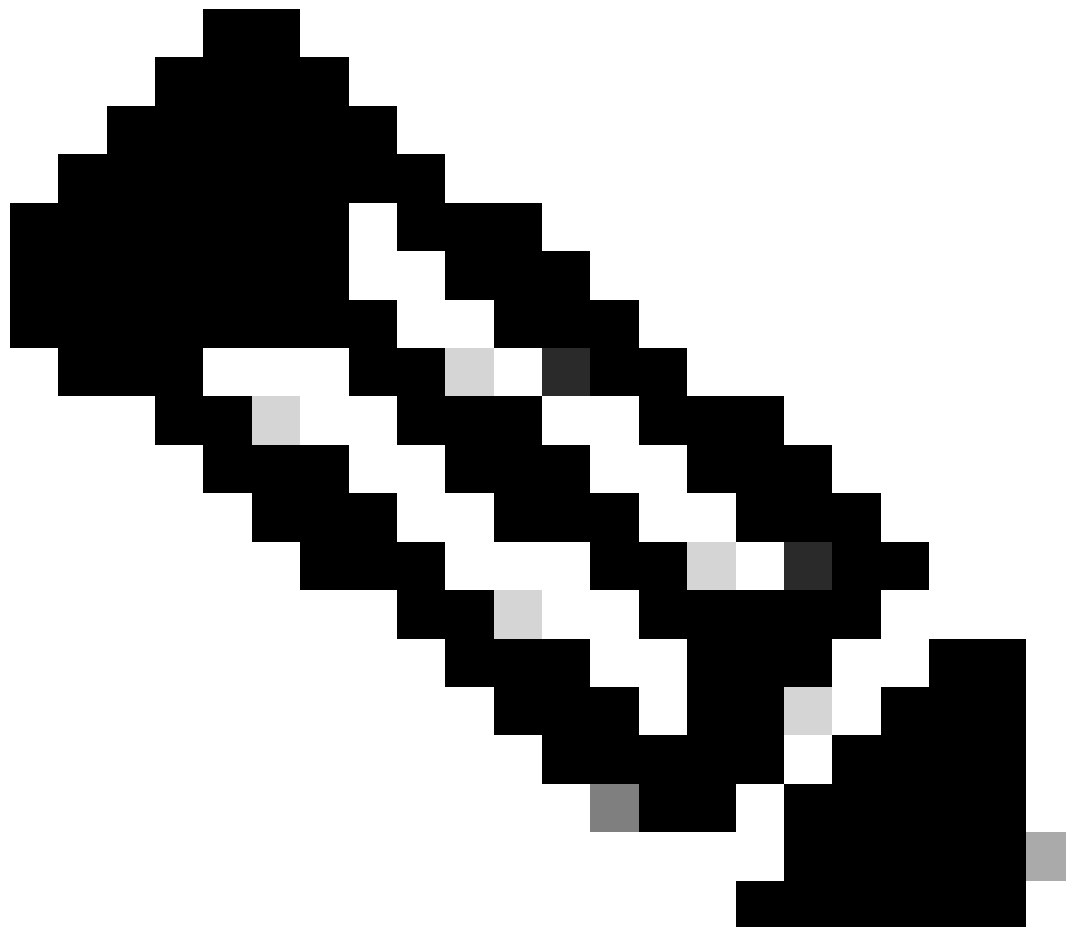
```
unit-3-1:*****
```

```
0 packet captured
```

```
0 packet shown
```

捕獲跟蹤

如果要檢視每台裝置上的資料平面如何處理入口資料包，請使用trace關鍵字。這會跟蹤前50個輸入資料包。您可以追蹤最多1000個輸入封包。



附註：如果在一個介面上應用了多個捕獲，則只能跟蹤單個資料包一次。

要跟蹤所有集群裝置上的介面OUTSIDE上的前1000個輸入資料包，請執行以下操作：

```
<#root>
firepower#
cluster exec cap CAPO int OUTSIDE buff 33554432 trace trace-count 1000 match tcp host 192.168.240.50 hos
```

捕獲興趣流後，需要確保跟蹤每個單元上的興趣包。需要記住的重要一點是，可以在裝置1-1上配置特定資料#1，但可以在其他裝置上#2，以此類推。

在本例中，您可以看到SYN/ACK是單元2-1上的資料包#2，但單元3-1上的資料包#1:

```
<#root>
firepower#
cluster exec show capture CAPO | include s.*ack

unit-1-1(LOCAL):*****
1: 12:58:31.117700 802.1Q vlan#202 PO 192.168.240.50.45468 > 192.168.241.50.80: S 441626016:441626016(0)
2: 12:58:31.118341 802.1Q vlan#202 PO 192.168.241.50.80 > 192.168.240.50.45468:

s

301658077:301658077(0)

ack

441626017 win 28960 <mss 1460,sackOK,timestamp 1125686319 1115330849,nop,wscale 7>

unit-2-1:*****

unit-3-1:*****
1: 12:58:31.111429 802.1Q vlan#202 PO 192.168.241.50.80 > 192.168.240.50.45468:

s

301658077:301658077(0)

ack

441626017 win 28960 <mss 1460,sackOK,timestamp 1125686319 1115330849,nop,wscale 7>
```

要在本地設#2上跟蹤資料包傳輸(SYN/ACK)，請執行以下操作：

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

```
cluster exec show cap CAPO packet-number 2 trace
```

```
unit-1-1(LOCAL):*****
```

```
2: 12:58:31.118341 802.1Q vlan#202 PO 192.168.241.50.80 > 192.168.240.50.45468:
```

```
s
```

```
301658077:301658077(0)
```

```
ack
```

```
441626017 win 28960 <mss 1460,sackOK,timestamp 1125686319 1115330849,nop,wscale 7>
```

```
Phase: 1
```

```
Type: CAPTURE
```

```
Subtype:
```

```
Result: ALLOW
```

```
Config:
```

```
Additional Information:
```

```
MAC Access list
```

```
...
```

要在遠端裝置上跟蹤同一資料包(SYN/ACK)，請執行以下操作：

```
<#root>
```

```
firepower#
```

```
cluster exec unit unit-3-1 show cap CAPO packet-number 1 trace
```

```
1: 12:58:31.111429 802.1Q vlan#202 PO 192.168.241.50.80 > 192.168.240.50.45468:
```

```
s
```

```
301658077:301658077(0)
```

```
ack
```

```
441626017 win 28960 <mss 1460,sackOK,timestamp 1125686319 1115330849,nop,wscale 7>
```

```
Phase: 1
```

```
Type: CAPTURE
```

```
Subtype:
```

```
Result: ALLOW
```

```
Config:
```

```
Additional Information:
```

```
MAC Access list
```

```
...
```

CCL擷取

要在CCL鏈路上啟用捕獲（在所有裝置上），請執行以下操作：

```
<#root>
```

```

firepower#
cluster exec capture CCL interface cluster

unit-1-1(LOCAL):*****
unit-2-1:*****
unit-3-1:*****

```

重新插入隱藏

預設情況下，在資料平面資料介面上啟用的捕獲會顯示所有資料包：

- 從物理網路到達的
- 從CCL重新注入的

如果您不想看到重新注入的資料包，請使用reinject-hide選項。如果您要驗證流是否不對稱，這將很有用：

```

<#root>
firepower#
cluster exec capture CAPI_RH reinject-hide interface INSIDE match tcp host 192.168.240.50 host 192.168.2

```

此捕獲僅顯示本地裝置在指定介面上直接從物理網路（而不是從其他集群裝置）收到的實際內容。

ASP刪除

如果要檢查特定流的軟體丟包，可以啟用asp-drop捕獲。如果您不知道要關注哪個丟棄原因，請使用關鍵字all。此外，如果您對資料包負載不感興趣，則可以指定headers-only 關鍵字。這樣可讓您擷取20-30倍以上的封包：

```

<#root>
firepower#
cluster exec cap ASP type asp-drop all buffer 33554432 headers-only

unit-1-1(LOCAL):*****
unit-2-1:*****
unit-3-1:*****

```

此外，還可以指定ASP捕獲中感興趣的IP:

```
<#root>
firepower#
cluster exec cap ASP type asp-drop all buffer 33554432 headers-only

match ip host 192.0.2.100 any
```

清除捕獲

清除在所有集群單元中運行的任何捕獲的緩衝區。這麼做不會停止擷取，但只會清除緩衝區：

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

unit-1-1(LOCAL):*****
unit-2-1:*****
unit-3-1:*****
```

停止捕獲

有兩種方法可以停止所有群集裝置上的活動捕獲。以後你可以繼續了。

方式1

```
<#root>
firepower#
cluster exec cap CAPI stop

unit-1-1(LOCAL):*****
unit-2-1:*****
unit-3-1:*****
```

恢復

```
<#root>
firepower#
cluster exec no capture CAPI stop
```

```
unit-1-1(LOCAL):*****
unit-2-1:*****
unit-3-1:*****
```

方式2

```
<#root>
firepower#
cluster exec no capture CAPI interface INSIDE

unit-1-1(LOCAL):*****
unit-2-1:*****
unit-3-1:*****
```

恢復

```
<#root>
firepower#
cluster exec capture CAPI interface INSIDE

unit-1-1(LOCAL):*****
unit-2-1:*****
unit-3-1:*****
```

收集捕獲

匯出捕獲的方法有多種。

方法1 — 到遠端伺服器

這樣您就可以將捕獲從資料平面上傳到遠端伺服器（例如TFTP）。捕獲名稱將自動更改以反映源裝置：

```
<#root>
firepower#
cluster exec copy /pcap capture:CAPI tftp://192.168.240.55/CAPI.pcap
```

unit-1-1(LOCAL):*****

Source capture name [CAPI]?

Address or name of remote host [192.168.240.55]?

Destination filename [CAPI.pcap]?

INFO: Destination filename is changed to unit-1-1_CAPI.pcap !!!!!!!

81 packets copied in 0.40 secs

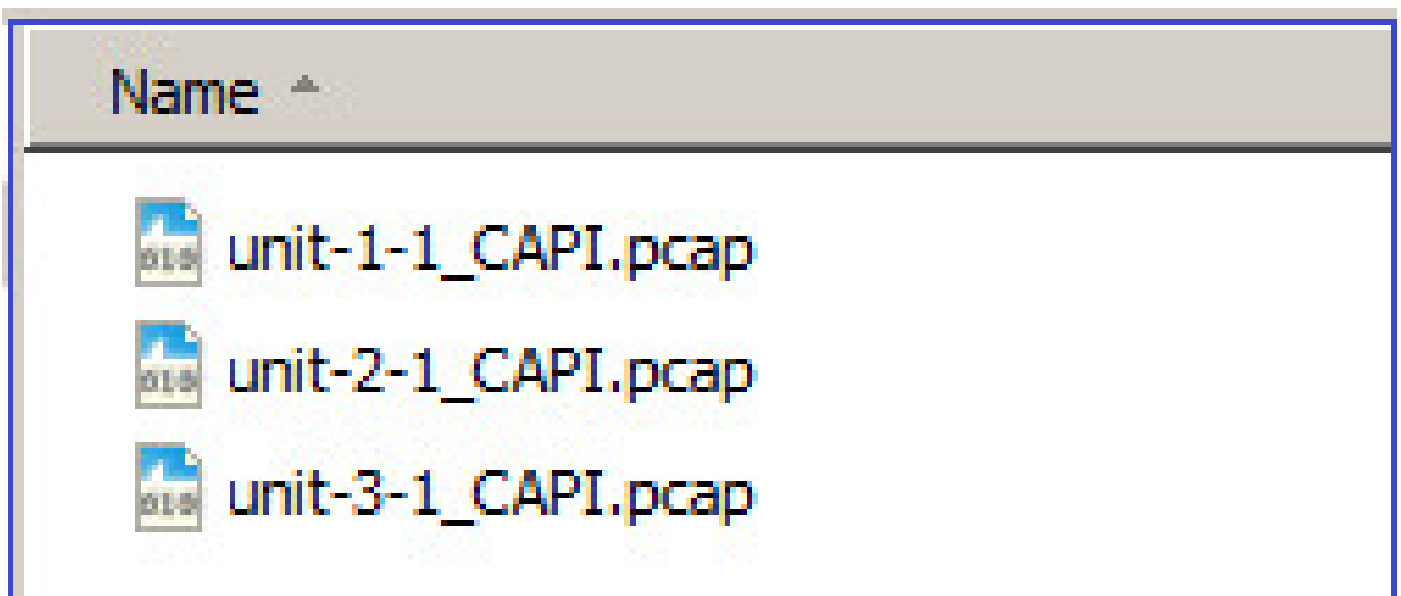
unit-2-1:*****

INFO: Destination filename is changed to unit-2-1_CAPI.pcap !

unit-3-1:*****

INFO: Destination filename is changed to unit-3-1_CAPI.pcap !

上載的pcap檔案：



方法2 — 從FMC取得擷取

這種方式僅適用於FTD。首先，將擷取複製到FTD磁碟上：

```
<#root>
```

```
firepower#
```

```
cluster exec copy /pcap capture:CAPI disk0:CAPI.pcap
```

```
unit-1-1(LOCAL):*****
```

```
Source capture name [CAPI]?
```

```
Destination filename [CAPI.pcap]?
```

```
!!!!
```

```
62 packets copied in 0.0 secs
```

在專家模式下，將檔案從/mnt/disk0/複製到/ngfw/var/common/目錄：

```
<#root>
```

```
>
```

```
expert
```

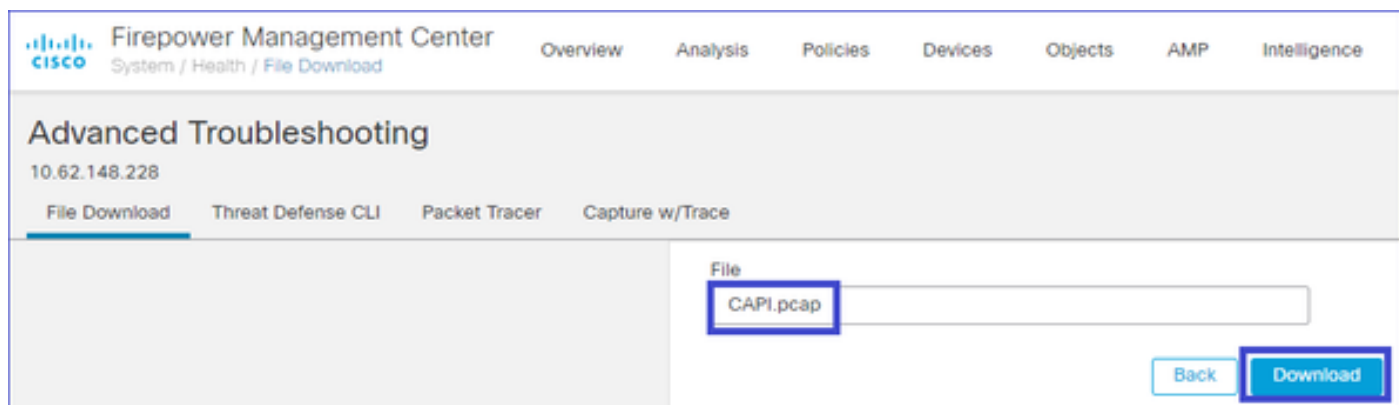
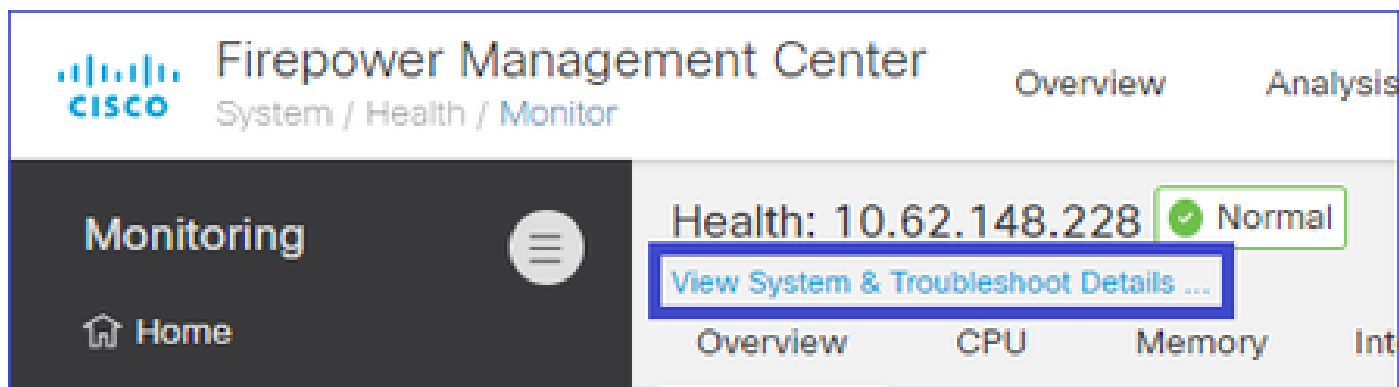
```
admin@firepower:~$
```

```
cd /mnt/disk0
```

```
admin@firepower:/mnt/disk0$
```

```
sudo cp CAPI.pcap /ngfw/var/common
```

最後，在FMC上導航到System > Health > Monitor部分。選擇View System & Troubleshoot Details > Advanced Troubleshooting並獲取捕獲檔案：



刪除捕獲

要從所有集群裝置刪除捕獲，請使用以下命令：

```
<#root>  
firepower#  
cluster exec no capture CAPI  
  
unit-1-1(LOCAL):*****  
unit-2-1:*****  
unit-3-1:*****
```

解除安裝的流

在FP41xx/FP9300上，可以靜態或動態地將資料流解除安裝到硬體加速器（例如Fastpath規則）。有關流量解除安裝的更多詳細資訊，請檢視以下文檔：

<https://www.cisco.com/c/en/us/support/docs/security/firepower-ngfw/212321-clarify-the-firepower-threat-defense-acc.html#anc22>

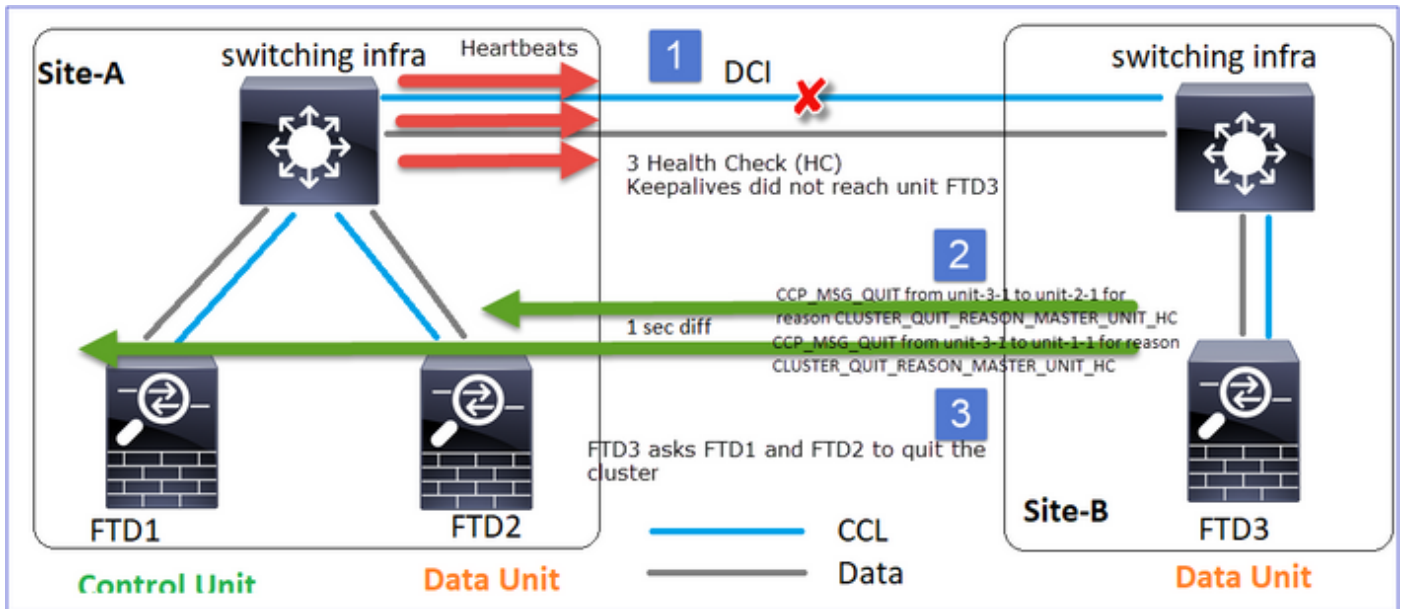
如果流量被分流，則只有少數封包會通過FTD資料平面。其餘部分由硬體加速器（智慧網絡卡）處理。

從捕獲的角度來看，這表示如果只啟用FTD資料平面級捕獲，您將看不到經過裝置的所有封包。在這種情況下，您還需要啟用FXOS機箱級捕獲。

集群控制鏈路(CCL)消息

如果您在CCL上進行捕獲，您會注意到集群裝置會交換不同型別的消息。人們感興趣的是：

通訊協定	說明
UDP通49495	<p>叢集心跳(keepalive)</p> <ul style="list-style-type: none">· L3廣播(255.255.255.255)· 每個集群裝置均按運行狀況檢查保持時間值的1/3傳送這些資料包。· 請注意，在擷取中看49495的所有UDP封包並非都是心跳· 心跳包含序列號。

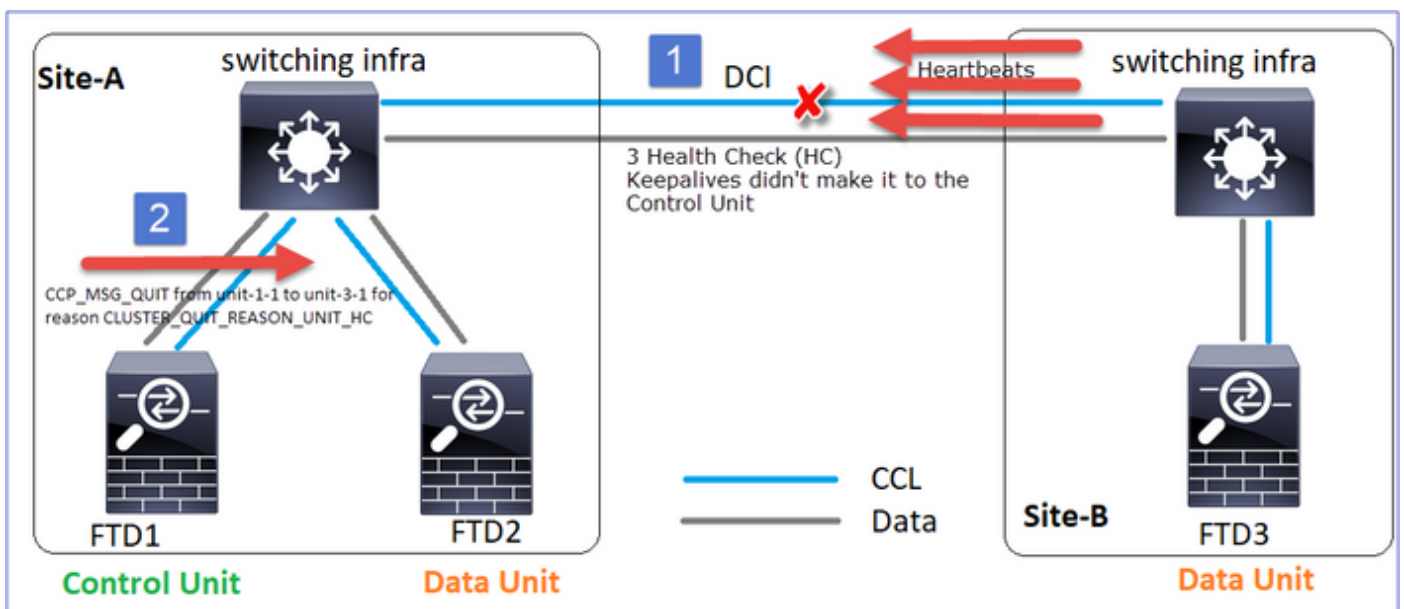


問：CLUSTER_QUIT_REASON_PRIMARY_UNIT_HC的作用是什麼？

A.從unit-3-1(Site-B)的角度來看，它丟失了站點A與unit-1-1和unit-2-1的連線，因此它需要儘快將其從成員清單中刪除，否則，如果unit-2-1仍在其成員清單中，並且unit-2-1恰好是連線的控制器，則對unit-2-1的流查詢失敗，則它可能會丟失資料包。

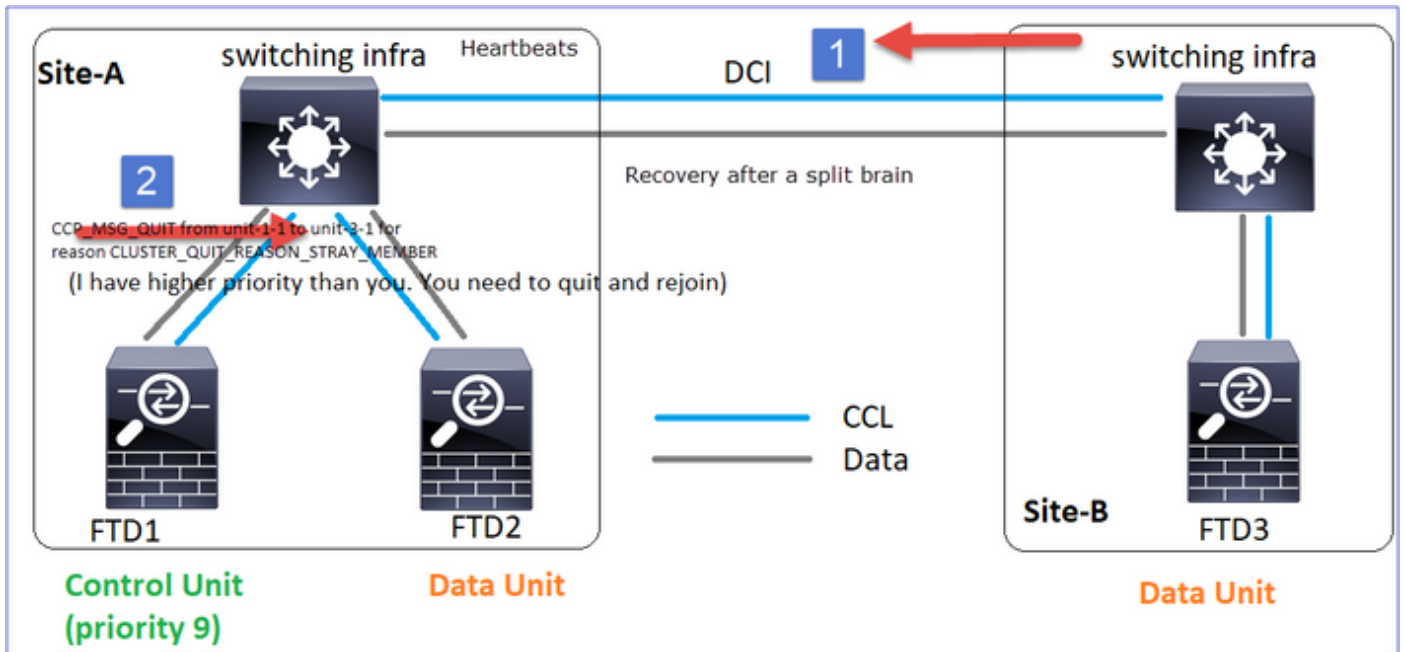
CLUSTER_QUIT_REASON_UNIT_HC

每當控制節點從資料節點丟失3個連續心跳消息時，它都會通過CCL傳送CLUSTER_QUIT_REASON_UNIT_HC消息。此消息為單播。



CLUSTER_QUIT_REASON_STRAY_MEMBER

當拆分分割槽與對等分割槽重新連線時，主控制單元將新資料節點視為雜散成員，並接收具有CLUSTER_QUIT_REASON_STRAY_MEMBER原因的CCP退出消息。



CLUSTER_QUIT_MEMBER_DROPPED

由資料節點生成並作為廣播傳送的廣播消息。裝置收到此消息後，將進入DISABLED狀態。此外，自動重新連線不會啟動：

```
<#root>
```

```
firepower#
```

```
show cluster info trace | include DROPOUT
```

```
Nov 04 00:22:54.699 [DEBUG]Receive CCP message: CCP_MSG_QUIT from unit-3-1 to unit-1-1 for reason
CLUSTER_QUIT_MEMBER_DROPOUT
```

```
Nov 04 00:22:53.699 [DEBUG]Receive CCP message: CCP_MSG_QUIT from unit-3-1 to unit-2-1 for reason
CLUSTER_QUIT_MEMBER_DROPOUT
```

集群歷史記錄顯示：

```
<#root>
```

```
PRIMARY          DISABLED          Received control message DISABLE (
member dropout announcement
)
```

叢集健康狀況檢查(HC)機制

主要重點：

- 每個集群單元每1/3的運行狀況檢查保持時間值傳送一次心跳（廣播255.255.255.255），並使用UDP埠49495作為CCL上的傳輸。
- 每個集群單元使用輪詢計時器和輪詢計數值獨立跟蹤其他單元。
- 如果集群單元在心跳間隔內沒有收到來自集群對等單元的任何資料包（心跳或資料包），它將增加輪詢計數值的值。
- 當集群對等裝置的輪詢計數值變為3時，對等裝置被視為關閉。
- 每當收到心跳時，都會檢查其序列號，並且在與先前收到的心跳的差異不同於1的情況下，心跳丟棄計數器也會相應地增加。
- 如果群集對等體的Poll count計數器不同於0，且對等體接收到資料包，則該計數器將重置為0值。

使用以下命令檢查群集運行狀況計數器：

```
<#root>
```

```
firepower#
```

```
show cluster info health details
```

Unit (ID)	Heartbeat count	Heartbeat drops	Average gap (ms)	Maximum slip (ms)	Poll count
unit-2-1 (1)	650	0	4999	1	0
unit-3-1 (2)	650	0	4999	1	0

主列的說明

列	說明
單位(ID)	遠端群集對等體的ID。
心跳計數	通過CCL從遠端對等裝置接收的心跳數。
心跳丟棄	丟失的心跳數。此計數器根據收到的心跳序列號來計算。
平均差距	收到的心跳的平均時間間隔。
輪詢計數	當此計數器變為3時，裝置將從群集中刪除。輪詢查詢間隔與心

	跳間隔相同，但獨立運行。
--	--------------

要重置計數器，請使用以下命令：

```
<#root>
```

```
firepower#
```

```
clear cluster info health details
```

如何驗證心跳頻率？

A.檢查平均差距值：

```
<#root>
```

```
firepower#
```

```
show cluster info health details
```

```
-----  
|                Unit (ID)| Heartbeat| Heartbeat|
```

```
Average
```

```
| Maximum|      Poll|  
|                | count|      drops|
```

```
gap (ms)
```

```
| slip (ms)|      count|
```

```
-----  
|                unit-2-1 ( 1)|      3036|          0|
```

```
999
```

```
|                1|          0|  
-----
```

問：如何更改FTD上的集群保持時間？

A.使用FlexConfig

在大腦分裂之後，誰成了控制節點？

A.具有最高優先順序（最低編號）的單位：

```
<#root>
```

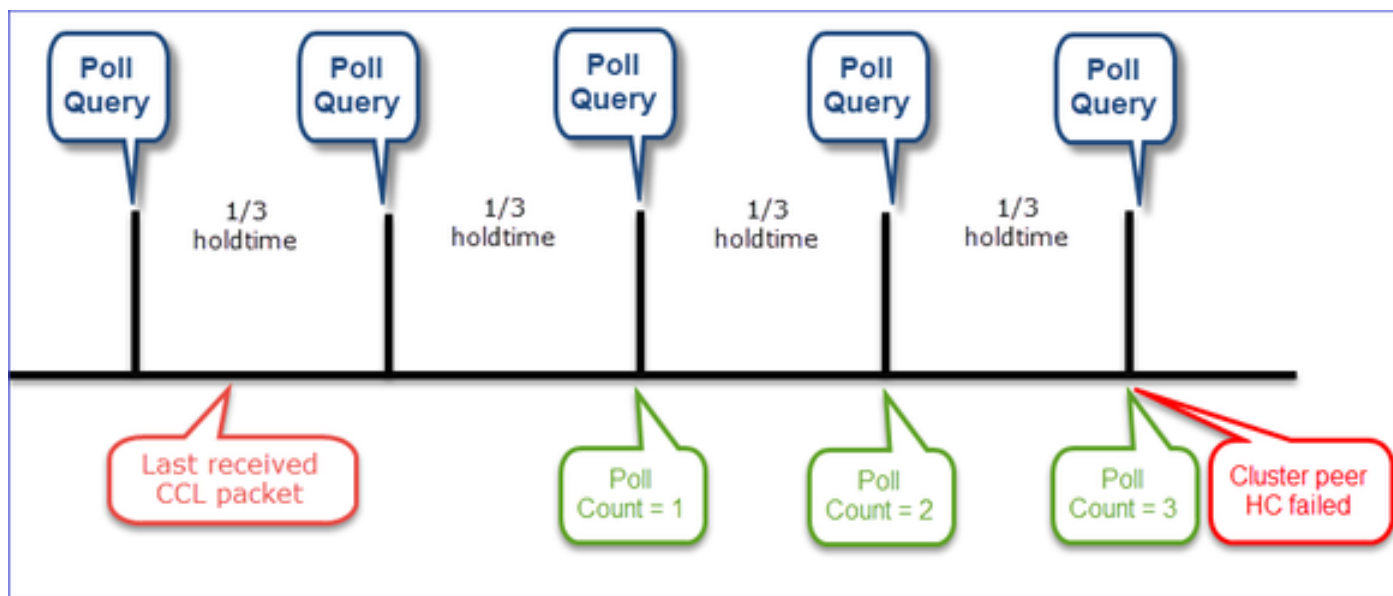
```
firepower#
```

show run cluster | include priority

priority 9

有關更多詳細資訊，請檢視HC故障場景1。

集群式HC機構視覺化



指示性計時器：最小和最大值取決於最後收到的CCL資料包到達。

保留時間	輪詢查詢檢查 (頻率)	最小檢測時間	最長檢測時間
3秒 (預設)	~1秒	~3.01秒	~3.99秒
4秒	~1.33秒	~4.01秒	~5.32秒
5秒	~1.66秒	~5.01秒	~6.65秒
6秒	約2秒	~6.01秒	~7.99秒
7秒	~2.33秒	~7.01秒	~9.32秒

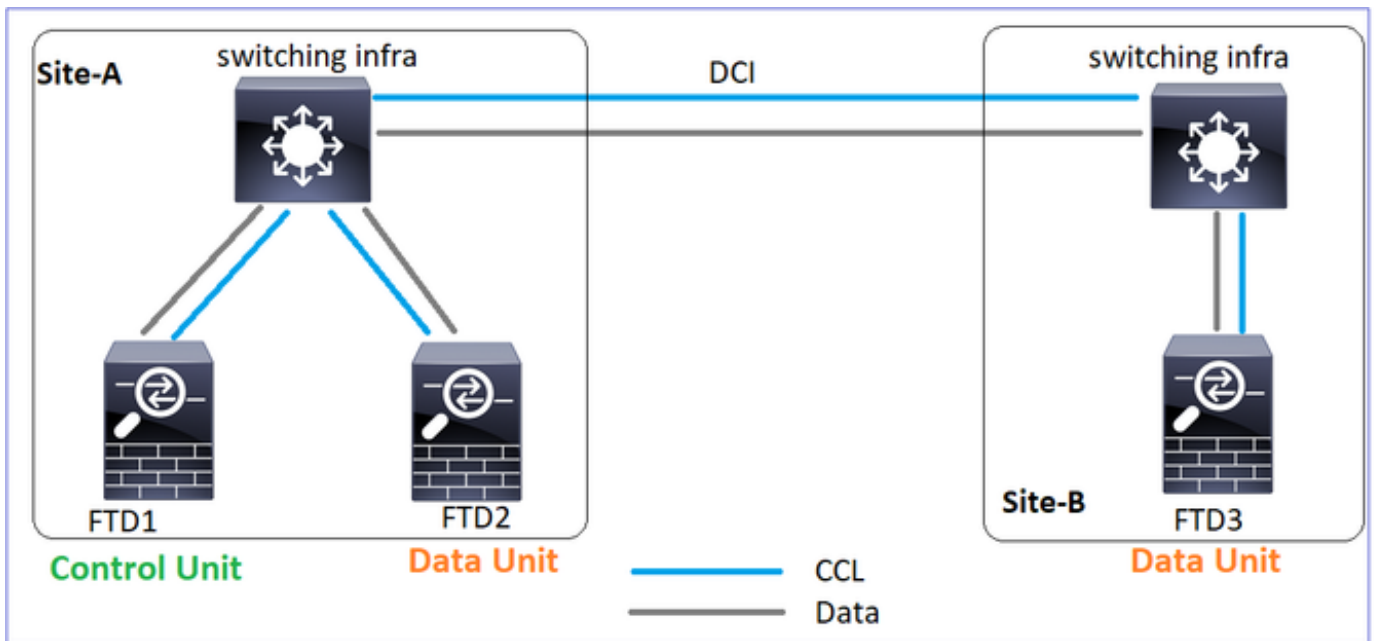
8秒	~2.66秒	~8.01秒	~10.65秒
----	--------	--------	---------

集群HC故障場景

本部分的目標是演示：

- 不同的群集HC故障場景。
- 不同日誌和命令輸出的關聯方式。

拓撲



群集配置

Unit-1-1	Unit-2-1
<pre> cluster group GROUP1 key ***** local-unit unit-1-1 cluster-interface Port-channel48 ip 10.17.1.1 255.255.0.0 priority 9 health-check holdtime 3 health-check data-interface auto-rejoin 3 5 2 health-check cluster-interface auto-rejoin unlimited 5 1 health-check system auto-rejoin 3 5 2 health-check monitor-interface debounce-time 500 site-id 1 enable </pre>	<pre> cluster group GROUP1 key ***** local-unit unit-2-1 cluster-interface Port-channel48 ip 10.17.1.2 255.255.0.0 priority 17 health-check holdtime 3 health-check data-interface auto-rejoin 3 5 2 health-check cluster-interface auto-rejoin unlimited 5 1 health-check system auto-rejoin 3 5 2 health-check monitor-interface debounce-time 500 site-id 1 enable </pre>

群集狀態

Unit-1-1	Unit-2-1
<pre><#root> firepower# show cluster info Cluster GROUP1: On Interface mode: spanned This is "unit-1-1" in state PRIMARY ID : 0 Site ID : 1 Version : 9.12(2)33 Serial No.: FCH22247LNK CCL IP : 10.17.1.1 CCL MAC : 0015.c500.018f Last join : 20:25:36 UTC Nov 1 2020 Last leave: 20:25:28 UTC Nov 1 2020 Other members in the cluster: Unit "unit-3-1" in state secondary ID : 1 Site ID : 2 Version : 9.12(2)33 Serial No.: FCH22247MKJ CCL IP : 10.17.3.1 CCL MAC : 0015.c500.038f Last join : 20:58:45 UTC Nov 1 2020 Last leave: 20:58:37 UTC Nov 1 2020 Unit "unit-2-1" in state SECONDARY ID : 2 Site ID : 1 Version : 9.12(2)33 Serial No.: FCH23157Y9N CCL IP : 10.17.2.1 CCL MAC : 0015.c500.028f Last join : 20:44:45 UTC Nov 1 2020 Last leave: 20:44:38 UTC Nov 1 2020</pre>	<pre><#root> firepower# show cluster info Cluster GROUP1: On Interface mode: spanned This is "unit-2-1" in state SECONDARY ID : 2 Site ID : 1 Version : 9.12(2)33 Serial No.: FCH23157Y9N CCL IP : 10.17.2.1 CCL MAC : 0015.c500.028f Last join : 20:44:46 UTC Nov 1 2020 Last leave: 20:44:38 UTC Nov 1 2020 Other members in the cluster: Unit "unit-1-1" in state PRIMARY ID : 0 Site ID : 1 Version : 9.12(2)33 Serial No.: FCH22247LNK CCL IP : 10.17.1.1 CCL MAC : 0015.c500.018f Last join : 20:25:36 UTC Nov 1 2020 Last leave: 20:25:28 UTC Nov 1 2020 Unit "unit-3-1" in state SECONDARY ID : 1 Site ID : 2 Version : 9.12(2)33 Serial No.: FCH22247MKJ CCL IP : 10.17.3.1 CCL MAC : 0015.c500.038f Last join : 20:58:45 UTC Nov 1 2020 Last leave: 20:58:37 UTC Nov 1 2020</pre>

案例 1

兩個方向的CCL通訊丟失超過4秒。

在失敗之前

FTD1	FTD2	FTD3
站點A	站點A	站點B
控制節點	資料節點	資料節點

恢復後 (裝置角色不變)

FTD1	FTD2	FTD3
站點A	站點A	站點B
控制節點	資料節點	資料節點

分析

故障 (CCL通訊丟失)。

The image shows three terminal windows from a Firepower device. The first window, titled 'unit-1-1 Control Unit', shows commands like 'clear cluster info trace' and 'clear cap /'. The second window, titled 'unit-2-1 Data Unit', shows a series of 'firepower#' prompts. The third window, titled 'unit-3-1 Data Unit', shows a warning about dynamic routing and a status change: 'Cluster unit unit-3-1 transitioned from SECONDARY to PRIMARY'.

Unit-3-1上的資料平面控制檯消息：

<#root>

firepower#

WARNING: dynamic routing is not supported on management interface when cluster interface-mode is 'spanned'. If dynamic routing is configured on any management interface, please remove it.

Cluster unit unit-3-1 transitioned from SECONDARY to PRIMARY

Cluster disable is performing cleanup..done.
 All data interfaces have been shutdown due to clustering being disabled.
 To recover either enable clustering or remove cluster group configuration.

Unit-1-1群集跟蹤日誌：

```
<#root>
```

```
firepower#
```

```
show cluster info trace | include unit-3-1
```

```
Nov 02 09:38:14.239 [INFO]Notify chassis de-bundle port for blade unit-3-1, stack 0x000055a8918307fb 0x
Nov 02 09:38:14.239 [INFO]FTD - CD proxy received state notification (DISABLED) from unit unit-3-1
Nov 02 09:38:14.239
```

```
[DEBUG]Send CCP message to all: CCP_MSG_QUIT from unit-1-1 to unit-3-1 for reason CLUSTER_QUIT_MEMBER_DR
```

```
Nov 02 09:38:14.239 [INFO]Notify chassis de-bundle port for blade unit-3-1, stack 0x000055a8917eb596 0x
Nov 02 09:38:14.239
```

```
[DEBUG]Send CCP message to id 1: CCP_MSG_QUIT from unit-1-1 to unit-3-1 for reason CLUSTER_QUIT_REASON_UN
```

```
Nov 02 09:38:14.239 [CRIT]Received heartbeat event 'SECONDARY heartbeat failure' for member unit-3-1 (I
```

裂腦

Unit-1-1	Unit-2-1
<pre><#root> firepower# show cluster info Cluster GROUP1: On Interface mode: spanned This is "unit-1-1" in state PRIMARY ID : 0 Site ID : 1 Version : 9.12(2)33 Serial No.: FCH22247LNK CCL IP : 10.17.1.1 CCL MAC : 0015.c500.018f Last join : 20:25:36 UTC Nov 1 2020</pre>	<pre><#root> firepower# show cluster info Cluster GROUP1: On Interface mode: spanned This is "unit-2-1" in state S ID : 2 Site ID : 1 Version : 9.12(2)33 Serial No.: FCH23157Y9N CCL IP : 10.17.2.1 CCL MAC : 0015.c500.028 Last join : 20:44:46 UTC Last leave: 20:44:38 UTC Other members in the cluster:</pre>

<pre> Last leave: 20:25:28 UTC Nov 1 2020 Other members in the cluster: Unit "unit-2-1" in state SECONDARY ID : 2 Site ID : 1 Version : 9.12(2)33 Serial No.: FCH23157Y9N CCL IP : 10.17.2.1 CCL MAC : 0015.c500.028f Last join : 20:44:45 UTC Nov 1 2020 Last leave: 20:44:38 UTC Nov 1 2020 </pre>	<pre> Unit "unit-1-1" in state PRIMARY ID : 0 Site ID : 1 Version : 9.12(2)33 Serial No.: FCH22247LNK CCL IP : 10.17.1.1 CCL MAC : 0015.c500.018 Last join : 20:25:36 UTC Last leave: 20:25:28 UTC </pre>
--	--

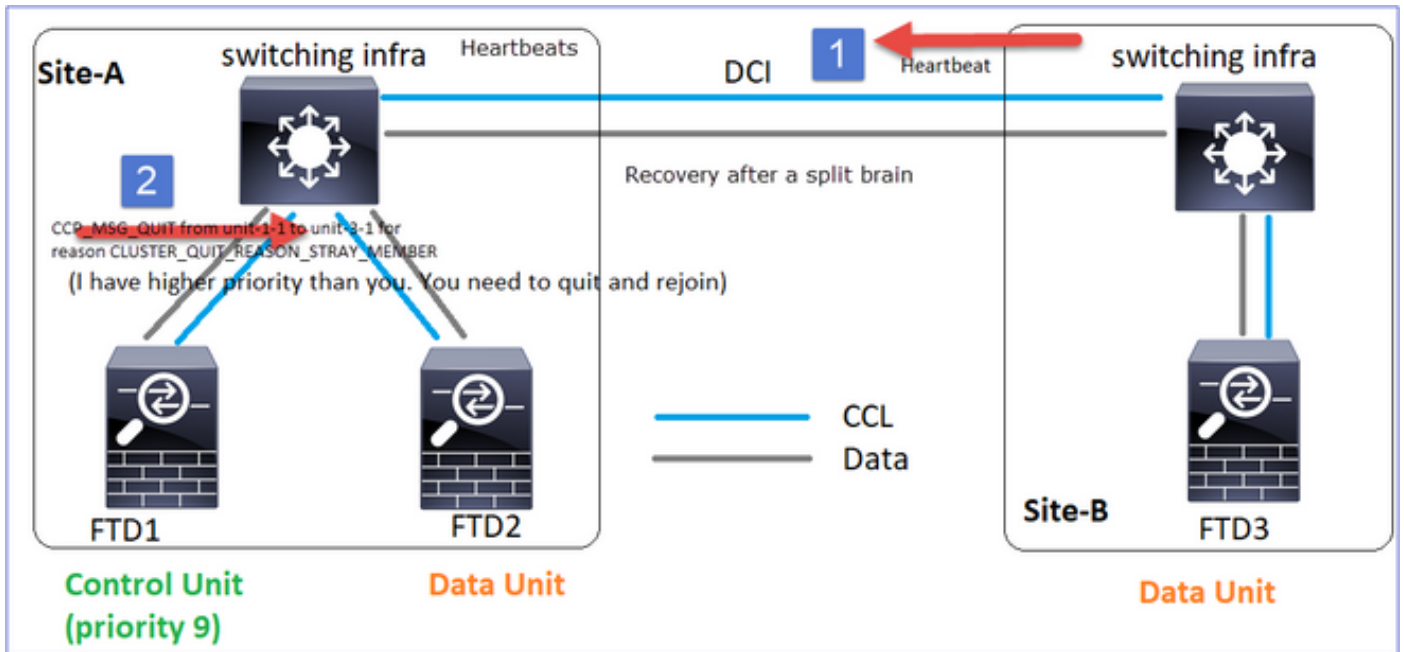
群集歷史記錄

Unit-1-1	Unit-2-1	Unit-3-1
無事件	無事件	<pre> <#root> 09:38:16 UTC Nov 2 2020 SECONDARY PRIMARY_POST_CONFIG Primary relinquished 09:38:17 UTC Nov 2 2020 PRIMARY_POST_CONFIG Primary Primary post config done </pre>

CCL通訊恢復

Unit-1-1檢測當前控制節點，並且由於unit-1-1的優先順序較高，因此會向unit-3-1傳送 CLUSTER_QUIT_REASON_STRAY_MEMBER消息以觸發新的選舉過程。最後，unit-3-1作為資料節點重新加入。

當拆分分割槽與對等分割槽重新連線時，資料節點被主控制節點視為雜散成員，並接收具有 CLUSTER_QUIT_REASON_STRAY_MEMBER原因的CCP退出消息。



<#root>

Unit-3-1 console logs show:

```
Cluster unit unit-3-1 transitioned from PRIMARY to DISABLED
```

The 3DES/AES algorithms require a Encryption-3DES-AES activation key.

```
Detected Cluster Primart.
```

```
Beginning configuration replication from Primary.
```

```
WARNING: Local user database is empty and there are still 'aaa' commands for 'LOCAL'.
```

```
..
```

```
Cryptochecksum (changed): a9ed686f 8e2e689c 2553a104 7a2bd33a
```

```
End configuration replication from Primary.
```

```
Cluster unit unit-3-1 transitioned from DISABLED to SECONDARY
```

兩個裝置 (unit-1-1和unit-3-1) 都在其集群日誌中顯示 :

<#root>

```
firepower#
```

```
show cluster info trace | include retain
```

```
Nov 03 21:20:23.019 [CRIT]Found a split cluster with both unit-1-1 and unit-3-1 as primary units. Prima
```

```
Nov 03 21:20:23.019 [CRIT]Found a split cluster with both unit-1-1 and unit-3-1 as primary units. Prima
```

還有為拆分大腦生成的syslog消息：

```
<#root>
```

```
firepower#
```

```
show log | include 747016
```

```
Nov 03 2020 21:20:23: %FTD-4-747016: Clustering: Found a split cluster with both unit-1-1 and unit-3-1
```

```
Nov 03 2020 21:20:23: %FTD-4-747016: Clustering: Found a split cluster with both unit-1-1 and unit-3-1
```

群集歷史記錄

Unit-1-1	Unit-2-1	Unit-3-1
無事件	無事件	<pre><#root> 09:47:33 UTC Nov 2 2020 Primary DISABLED Detected a splitted cluster 09:47:38 UTC Nov 2 2020 DISABLED ELECTION Enabled from CLI 09:47:38 UTC Nov 2 2020 ELECTION SECONDARY_COLD Received cluster control me 09:47:38 UTC Nov 2 2020 SECONDARY_COLD SECONDARY_APP_SYNC Client progression done 09:48:18 UTC Nov 2 2020 SECONDARY_APP_SYNC SECONDARY_CONFIG SECONDARY application c 09:48:29 UTC Nov 2 2020 SECONDARY_CONFIG SECONDARY_FILESYS Configuration replicati 09:48:30 UTC Nov 2 2020 SECONDARY_FILESYS SECONDARY_BULK_SYNC Client progression done 09:48:54 UTC Nov 2 2020 SECONDARY_BULK_SYNC SECONDARY Client progression done</pre>

案例 2

兩個方向的CCL通訊丟失大約3-4秒。

在失敗之前

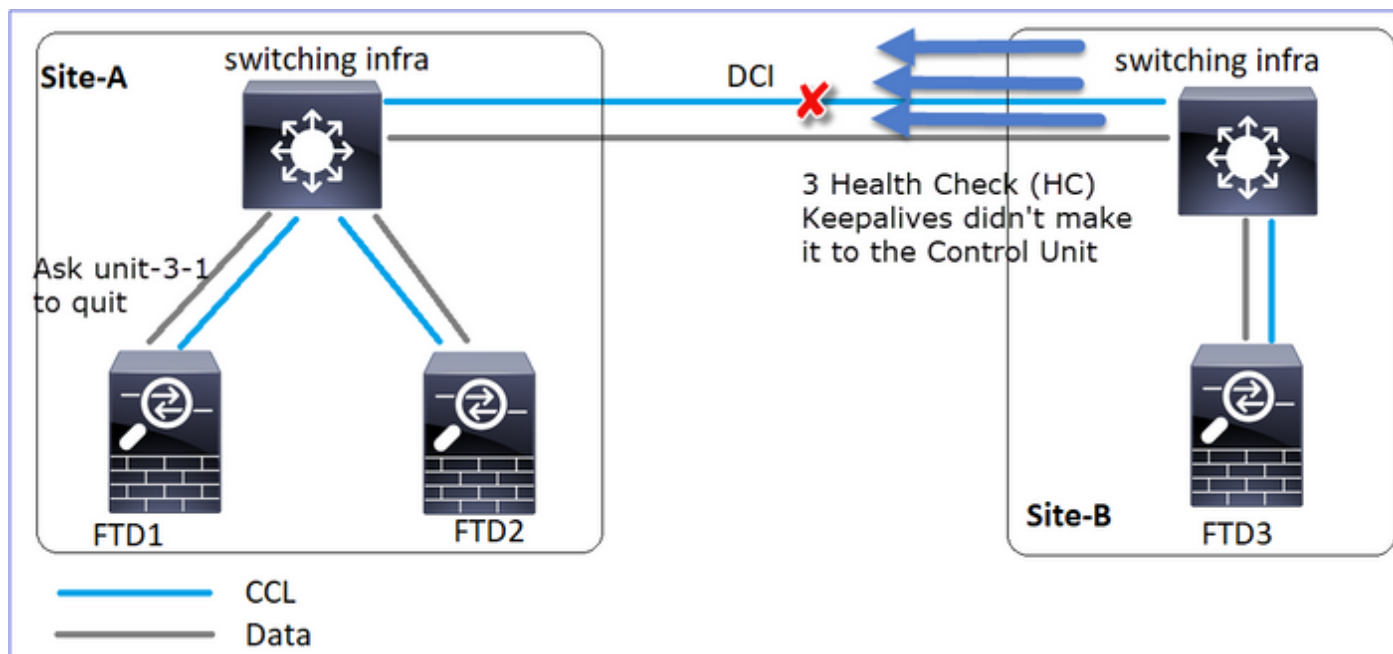
FTD1	FTD2	FTD3
站點A	站點A	站點B
控制節點	資料節點	資料節點

恢復後 (裝置角色不變)

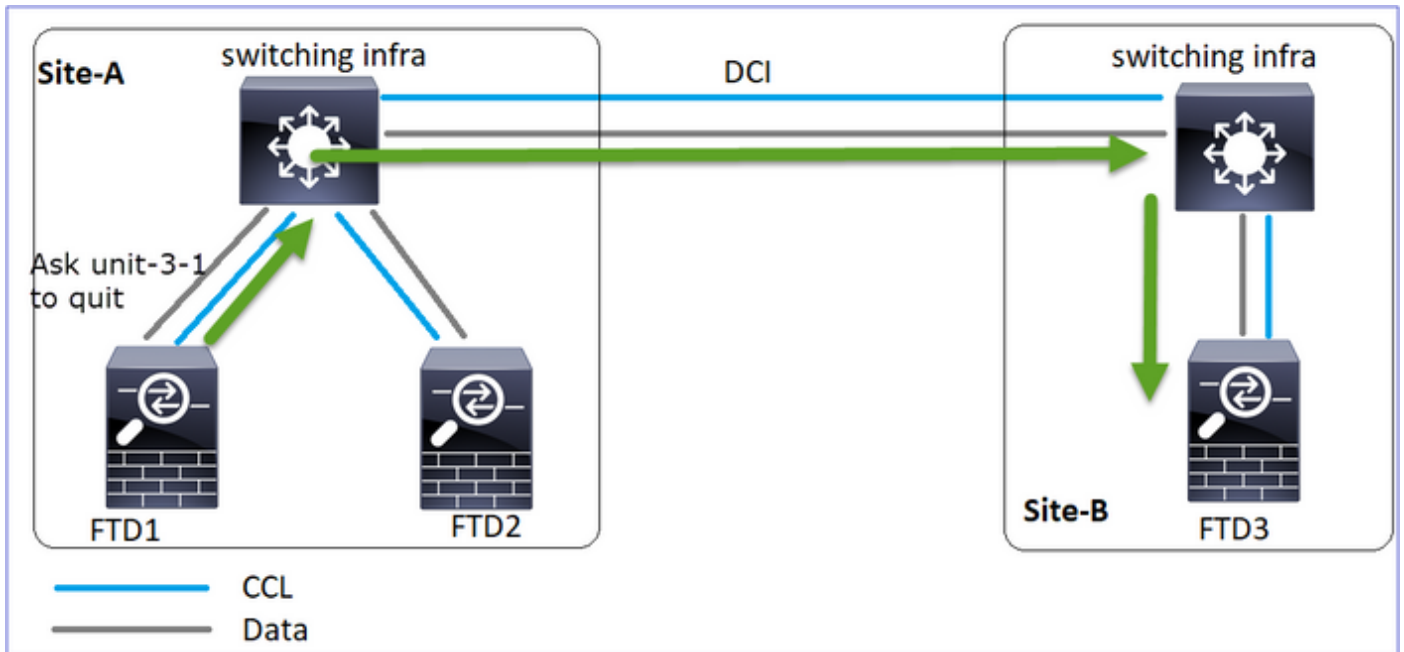
FTD1	FTD2	FTD3
站點A	站點A	站點B
控制節點	資料節點	資料節點

分析

活動1:控制節點從裝置3-1丟失3個HC，並向裝置3-1傳送消息以離開集群。



活動2:CCL恢復非常快，來自控制節點的CLUSTER_QUIT_REASON_STRAY_MEMBER消息到達了遠端端。Unit-3-1直接進入DISABLED模式，並且沒有拆分大腦



在unit-1-1(control)上，您會看到：

```
<#root>
```

```
firepower#
```

```
Asking SECONDARY unit unit-3-1 to quit because it failed unit health-check.
```

```
Forcing stray member unit-3-1 to leave the cluster
```

在unit-3-1(data node)上，您會看到：

```
<#root>
```

```
firepower#
```

```
Cluster disable
```

```
is performing cleanup..done.
```

```
All data interfaces have been shutdown due to clustering being disabled. To recover either enable cluster
```

```
Cluster unit unit-3-1 transitioned from SECONDARY to DISABLED
```

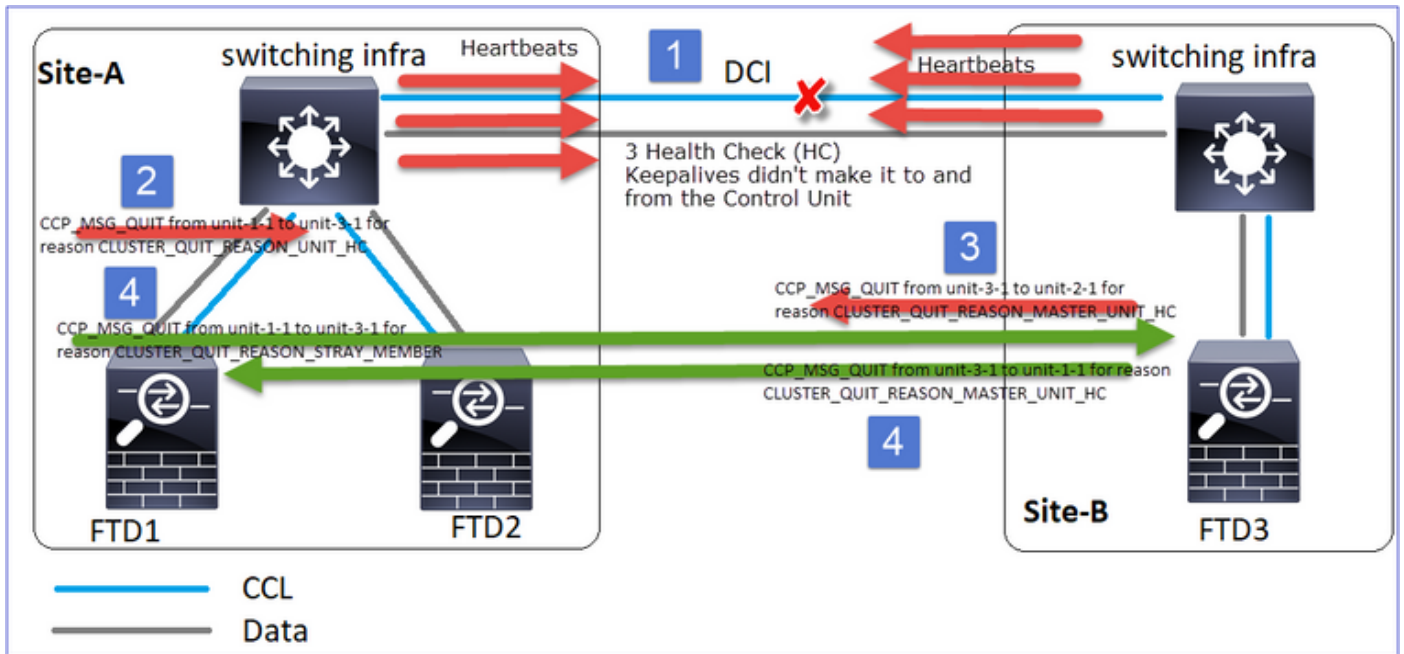
群集單元unit-3-1轉換到DISABLED狀態，一旦恢復CCL通訊，它就重新加入為資料節點：

```
<#root>
```

```
firepower#
```

```
show cluster history
```


分析



1. CCL關閉。
2. Unit-1-1不會從unit-3-1獲得3條HC消息，而是向unit-3-1傳送QUIT消息。此消息永遠無法到達unit-3-1。
3. Unit-3-1向unit-2-1傳送QUIT消息。此消息永遠無法到達unit-2-1。

CCL恢復。

4. Unit-1-1看到unit-3-1將自己通告為控制節點，並將QUIT_REASON_STRAY_MEMBER消息傳送到unit-3-1。一旦裝置-3-1獲得此消息進入DISABLED狀態。同時，Unit-3-1向Unit-1-1傳送QUIT_REASON_PRIMARY_UNIT_HC消息並請求其退出。一旦裝置1-1獲取此消息進入DISABLED狀態。

群集歷史記錄

```

Unit-1-1
<#root>
19:53:09 UTC Nov 2 2020
PRIMARY DISABLED
    Received control message DISABLE
                                (primary unit health check failure)
19:53:13 UTC Nov 2 2020
DISABLED ELECTION Enabled from CLI
19:53:13 UTC Nov 2 2020
ELECTION SECONDARY_COLD Received cluster control message
19:53:13 UTC Nov 2 2020
SECONDARY_COLD SECONDARY_APP_SYNC Client progression done
    
```

```

19:54:01 UTC Nov 2 2020
SECONDARY_APP_SYNC      SECONDARY_CONFIG      SECONDARY application configur
19:54:12 UTC Nov 2 2020
SECONDARY_CONFIG       SECONDARY_FILESYS     Configuration replication fini
19:54:13 UTC Nov 2 2020
SECONDARY_FILESYS      SECONDARY_BULK_SYNC   Client progression done
19:54:37 UTC Nov 2 2020
SECONDARY_BULK_SYNC

```

SECONDARY

Client progression done

案例 4

大約3-4秒的CCL通訊丟失

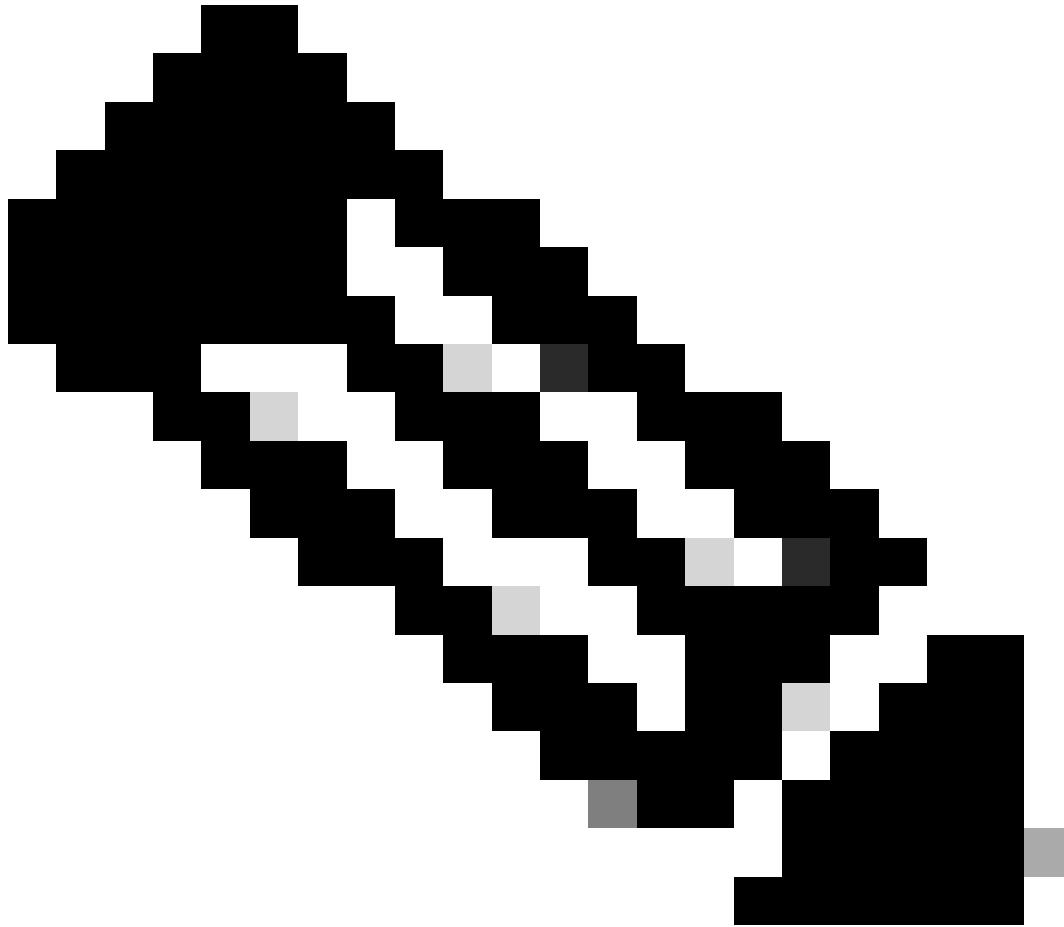
在失敗之前

FTD1	FTD2	FTD3
站點A	站點A	站點B
控制節點	資料節點	資料節點

恢復後 (控制節點更改了站點)

FTD1	FTD2	FTD3
------	------	------

- 出集群。CCL恢復。
5. Units-1-1和2-1作為資料節點重新加入集群。
-



附註：如果在步驟5中CCL未恢復，則在site-A中，FTD1會成為新的控制節點，並在CCL恢復之後，贏得新的選擇。

Unit-1-1上的Syslog消息：

```
<#root>
```

```
firepower#
```

```
show log | include 747
```

```
Nov 03 2020 23:13:08: %FTD-7-747005: Clustering: State machine notify event CLUSTER_EVENT_MEMBER_STATE
Nov 03 2020 23:13:09: %FTD-4-747015: Clustering: Forcing stray member unit-3-1 to leave the cluster
Nov 03 2020 23:13:09: %FTD-7-747005: Clustering: State machine notify event CLUSTER_EVENT_MEMBER_STATE
Nov 03 2020 23:13:10: %FTD-4-747015: Clustering: Forcing stray member unit-3-1 to leave the cluster
Nov 03 2020 23:13:10: %FTD-6-747004: Clustering:
```

State machine changed from state PRIMARY to DISABLED

```
Nov 03 2020 23:13:12: %FTD-7-747006: Clustering: State machine is at state DISABLED
Nov 03 2020 23:13:12: %FTD-7-747005: Clustering: State machine notify event CLUSTER_EVENT_MY_STATE (sta
Nov 03 2020 23:13:18: %FTD-6-747004: Clustering: State machine changed from state ELECTION to ONCALL
```

Unit-1-1上的群集跟蹤日誌：

<#root>

firepower#

show cluster info trace | include QUIT

```
Nov 03 23:13:10.789 [DEBUG]Send CCP message to all: CCP_MSG_QUIT from unit-1-1 for reason CLUSTER_QUIT_R
Nov 03 23:13:10.769 [DEBUG]
```

Receive CCP message: CCP_MSG_QUIT from unit-3-1 to unit-1-1 for reason CLUSTER_QUIT_REASON_PRIMARY_UNIT

```
Nov 03 23:13:10.769 [DEBUG]Send CCP message to id 1: CCP_MSG_QUIT from unit-1-1 to unit-3-1 for reason C
Nov 03 23:13:09.789 [DEBUG]Receive CCP message: CCP_MSG_QUIT from unit-2-1 for reason CLUSTER_QUIT_REASO
Nov 03 23:13:09.769 [DEBUG]Send CCP message to id 1: CCP_MSG_QUIT from unit-1-1 to unit-3-1 for reason C
Nov 03 23:13:08.559 [DEBUG]Send CCP message to all: CCP_MSG_QUIT from unit-1-1 to unit-3-1 for reason CL
Nov 03 23:13:08.559 [DEBUG]Send CCP message to id 1: CCP_MSG_QUIT from unit-1-1 to unit-3-1 for reason C
```

Unit-3-1上的Syslog消息：

<#root>

firepower#

show log | include 747

```
Nov 03 2020 23:13:09: %FTD-7-747005: Clustering: State machine notify event CLUSTER_EVENT_MEMBER_STATE
Nov 03 2020 23:13:10: %FTD-7-747005: Clustering: State machine notify event CLUSTER_EVENT_MEMBER_STATE
Nov 03 2020 23:13:10: %FTD-6-747004: Clustering:
```

State machine changed from state SECONDARY to PRIMARY

```
Nov 03 2020 23:13:10: %FTD-6-747004: Clustering: State machine changed from state PRIMARY_FAST to PRIMA
Nov 03 2020 23:13:10: %FTD-6-747004: Clustering: State machine changed from state PRIMARY_DRAIN to PRIM
Nov 03 2020 23:13:10: %FTD-6-747004: Clustering: State machine changed from state PRIMARY_CONFIG to PRI
Nov 03 2020 23:13:10: %FTD-7-747006: Clustering: State machine is at state PRIMARY_POST_CONFIG
Nov 03 2020 23:13:10: %FTD-6-747004: Clustering: State machine changed from state PRIMARY_POST_CONFIG t
Nov 03 2020 23:13:10: %FTD-7-747006: Clustering:
```

State machine is at state PRIMARY

群集歷史記錄

Unit-1-1

<#root>

23:13:13 UTC Nov 3 2020

PRIMARY DISABLED Received control message DISABLE
(primary unit health check failure)

23:13:18 UTC Nov 3 2020

DISABLED ELECTION Enabled from CLI

23:13:18 UTC Nov 3 2020

ELECTION ONCALL Received cluster control message

23:13:23 UTC Nov 3 2020

ONCALL ELECTION Received cluster control message

...
23:14:48 UTC Nov 3 2020
ONCALL ELECTION Received cluster control message

23:14:48 UTC Nov 3 2020
ELECTION SECONDARY_COLD Received cluster control message

23:14:48 UTC Nov 3 2020
SECONDARY_COLD SECONDARY_APP_SYNC Client progression done

23:15:36 UTC Nov 3 2020
SECONDARY_APP_SYNC SECONDARY_CONFIG SECONDARY application configuration
sync done

23:15:48 UTC Nov 3 2020
SECONDARY_CONFIG SECONDARY_FILESYS Configuration replication finished

23:15:49 UTC Nov 3 2020
SECONDARY_FILESYS SECONDARY_BULK_SYNC Client progression done

23:16:13 UTC Nov 3 2020
SECONDARY_BULK_SYNC

SECONDARY

Client progression done

案例 5

在失敗之前

FTD1	FTD2	FTD3
站點A	站點A	站點B
控制節點	資料節點	資料節點

Nov 04 00:51:47.019 [DEBUG]Send CCP message to all: CCP_MSG_QUIT from unit-2-1 for reason CLUSTER_QUIT_R
Nov 04 00:51:46.999 [DEBUG]

Receive CCP message: CCP_MSG_QUIT from unit-3-1 to unit-2-1 for reason CLUSTER_QUIT_REASON_PRIMARY_UNIT

Nov 04 00:51:45.389 [DEBUG]Receive CCP message: CCP_MSG_QUIT from unit-1-1 to unit-3-1 for reason CLUSTER

群集歷史記錄

Unit-1-1	Unit-2-1
無事件	<pre><#root> 00:51:50 UTC Nov 4 2020 SECONDARY DISABLED Received control message DISABLE (primary unit health check failure) 00:51:54 UTC Nov 4 2020 DISABLED ELECTION Enabled from CLI 00:51:54 UTC Nov 4 2020 ELECTION SECONDARY_COLD Received cluster control message 00:51:54 UTC Nov 4 2020 SECONDARY_COLD SECONDARY_APP_SYNC Client progression done 00:52:42 UTC Nov 4 2020 SECONDARY_APP_SYNC SECONDARY_CONFIG SECONDARY application configura sync done 00:52:54 UTC Nov 4 2020 SECONDARY_CONFIG SECONDARY_FILESYS Configuration replication finis 00:52:55 UTC Nov 4 2020 SECONDARY_FILESYS SECONDARY_BULK_SYNC Client progression done 00:53:19 UTC Nov 4 2020 SECONDARY_BULK_SYNC SECONDARY Client progression done</pre>

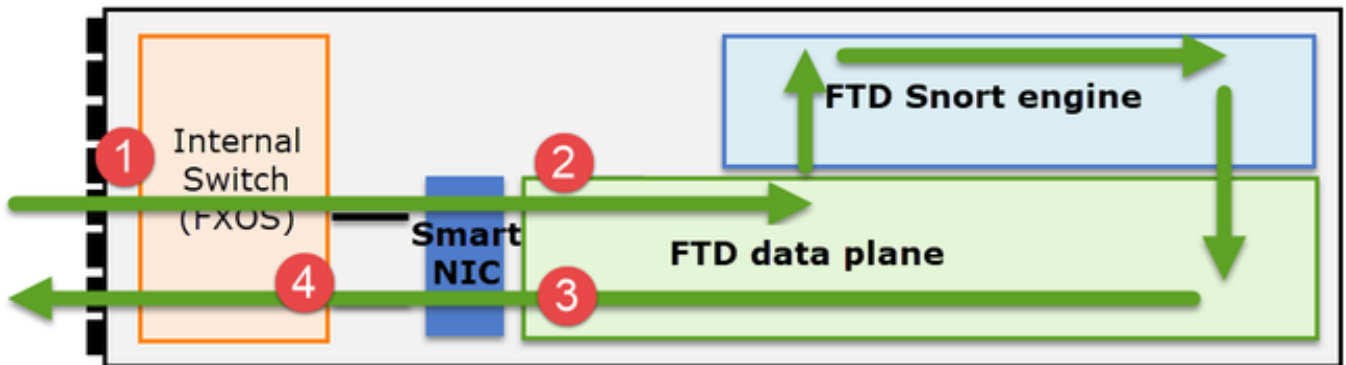
建立群集資料平面連線

NGFW擷取點

NGFW在這些點上提供捕獲功能：

- 機箱內部交換機(FXOS)
- FTD資料平面引擎
- FTD Snort引擎

當您對群集上的資料路徑問題進行故障排除時，大多數情況下使用的捕獲點是FXOS和FTD資料平面引擎捕獲。



1. 物理介面上的FXOS入口捕獲
2. 資料平面引擎中的FTD輸入擷取
3. 資料平面引擎中的FTD輸出擷取
4. 背板介面上的FXOS輸入擷取

有關NGFW捕獲的其他詳細資訊，請檢視以下文檔：

集群裝置流角色基礎知識

可通過多種方式通過集群建立連線，具體取決於以下因素：

- 流量型別 (TCP、UDP等)
- 在相鄰交換機上配置的負載均衡演算法
- 防火牆上配置的功能
- 網路條件 (例如IP分段、網路延遲等)

流角色	說明	標誌
所有者	通常，最初接收連線的裝置	UIO
主管	處理轉發器的所有者查詢請求的單元。	Y

備份所有者	只要控制器與所有者不是同一裝置，則控制器也是備份所有者。如果所有者選擇自己作為指揮交換機，則會選擇單獨的備份所有者。	Y (如果指揮交換機也是備份所有者) y (如果指揮交換機不是備份所有者)
轉發器	將資料包轉發給所有者的裝置	z
片段所有者	處理分段流量的單元	-
機箱備份	在機箱間集群中，當控制器/備份和所有者流都由同一機箱的單元擁有時，另一個機箱中的一個單元成為輔助備份/控制器。 此角色特定於具有1個以上刀片的 Firepower 9300系列的機箱間集群。	w

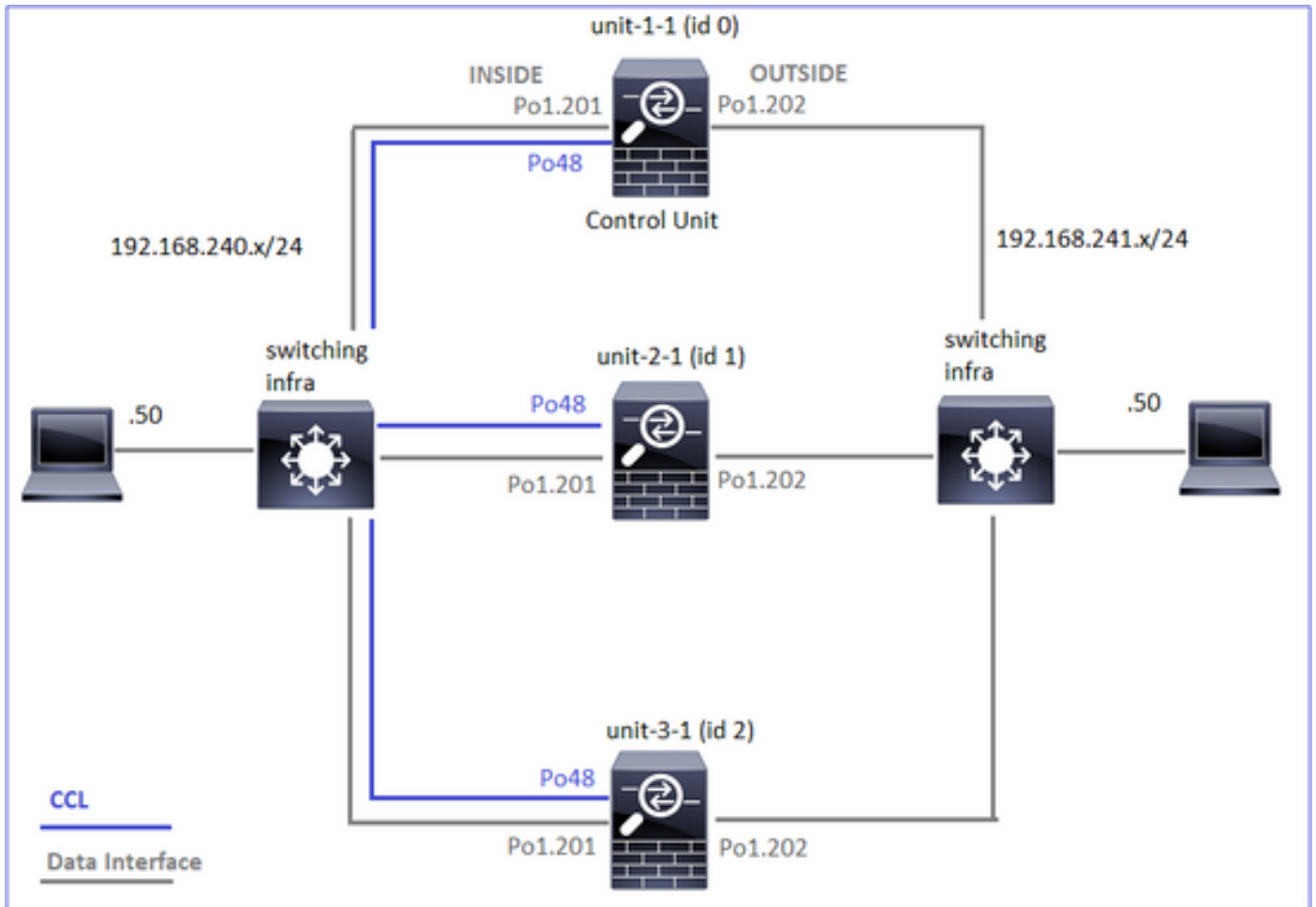
- 有關其他詳細資訊，請檢視《配置指南》中的相關章節 (參見相關資訊中的連結)
- 在特定情況下 (請參閱案例分析部分) ，某些標誌並不總是顯示。

集群連線建立案例研究

下一節將介紹各種案例研究，這些研究演示了通過群集建立連線的一些方法。其目標是：

- 熟悉不同的裝置角色。
- 演示如何關聯各種命令輸出。

拓撲




集群裝置和ID:

Unit-1-1	Unit-2-1
<pre> <#root> Cluster GROUP1: On Interface mode: spanned This is "unit-1-1" in state PRIMARY ID : 0 Site ID : 1 Version : 9.15(1) Serial No.: FCH22247LNK CCL IP : 10.17.1.1 CCL MAC : 0015.c500.018f Last join : 02:24:43 UTC Nov 27 2020 Last leave: N/A </pre>	<pre> <#root> Unit "unit-2-1" in state SECO ID : 1 Site ID : 1 Version : 9.15(1) Serial No.: FCH23157Y9N CCL IP : 10.17.2.1 CCL MAC : 0015.c500.02 Last join : 02:04:19 UTC Last leave: N/A </pre>

已啟用群集捕獲：

```
cluster exec cap CAPI int INSIDE buffer 33554432 match tcp host 192.168.240.50 host 192.168.241.50 eq 80
cluster exec cap CAPO int OUTSIDE buffer 33554432 match tcp host 192.168.240.50 host 192.168.241.50 eq 80
cluster exec cap CAPI_RH reinject-hide int INSIDE buffer 33554432 match tcp host 192.168.240.50 host 192.168.241.50 eq 80
cluster exec cap CAPO_RH reinject-hide int OUTSIDE buffer 33554432 match tcp host 192.168.240.50 host 192.168.241.50 eq 80
cluster exec cap CCL int cluster buffer 33554432
```

 附註：這些測試在實驗室環境中運行，通過群集的流量極小。在生產中，儘量使用特定的擷取過濾器（例如目的地連線埠和儘可能使用來源連線埠）來將擷取中的「噪音」降至最低。

案例研究1.對稱流量（所有者也是主管）

觀察1. reinject-hide捕獲僅顯示unit-1-1上的資料包。這意味著兩個方向的流都通過unit-1-1（對稱流量）：

```
<#root>
```

```
firepower#
```

```
cluster exec show cap
```

```
unit-1-1(LOCAL):*****
capture CCL type raw-data interface cluster [Capturing - 33513 bytes]
capture CAPI type raw-data buffer 33554432 trace interface INSIDE [Buffer Full - 33553914 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq 80
capture CAPO type raw-data buffer 33554432 trace interface OUTSIDE [Buffer Full - 33553914 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq 80
capture CAPI_RH type raw-data

reinject-hide

  buffer 33554432 interface INSIDE [Buffer Full] -
33553914 bytes
]
match tcp host 192.168.240.50 host 192.168.241.50 eq 80
capture CAPO_RH type raw-data

reinject-hide

  buffer 33554432 interface OUTSIDE [Buffer Full] -
33553914 bytes
]
match tcp host 192.168.240.50 host 192.168.241.50 eq 80

unit-2-1:*****
capture CCL type raw-data interface cluster [Capturing - 23245 bytes]
capture CAPI type raw-data buffer 33554432 trace interface INSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq 80
capture CAPO type raw-data buffer 33554432 trace interface OUTSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq 80
```

```

capture CAPI_RH type raw-data
reinject-hide

  buffer 33554432 interface INSIDE [Capturing -
0 bytes
]
match tcp host 192.168.240.50 host 192.168.241.50 eq 80
capture CAPO_RH type raw-data
reinject-hide

  buffer 33554432 interface OUTSIDE [Capturing -
0 bytes
]
match tcp host 192.168.240.50 host 192.168.241.50 eq 80

unit-3-1:*****
capture CCL type raw-data interface cluster [Capturing - 24815 bytes]
capture CAPI type raw-data buffer 33554432 trace interface INSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq 80
capture CAPO type raw-data buffer 33554432 trace interface OUTSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq 80
capture CAPI_RH type raw-data
reinject-hide

  buffer 33554432 interface INSIDE [Capturing -
0 bytes
]
match tcp host 192.168.240.50 host 192.168.241.50 eq 80
capture CAPO_RH type raw-data
reinject-hide

  buffer 33554432 interface OUTSIDE [Capturing -
0 bytes
]
match tcp host 192.168.240.50 host 192.168.241.50 eq 80

```

觀察2.源埠流量連線標誌分45954

<#root>

firepower#

cluster exec show conn

```

unit-1-1(LOCAL):*****
22 in use, 25 most used
Cluster:
fwd connections: 0 in use, 1 most used
dir connections: 0 in use, 122 most used
centralized connections: 0 in use, 0 most used

```

```

VPN redirect connections: 0 in use, 0 most used
Inspect Snort:
preserve-connection: 1 enabled, 0 in effect, 2 most enabled, 1 most in effect

TCP OUTSIDE 192.168.241.50:80 INSIDE 192.168.240.50:
45954
, idle 0:00:00, bytes 487413076,
flags UIO N1

```

```

unit-2-1:*****
22 in use, 271 most used
Cluster:
fwd connections: 0 in use, 2 most used
dir connections: 0 in use, 2 most used
centralized connections: 0 in use, 0 most used
VPN redirect connections: 0 in use, 0 most used
Inspect Snort:
preserve-connection: 1 enabled, 0 in effect, 249 most enabled, 0 most in effect

```

```

unit-3-1:*****
17 in use, 20 most used
Cluster:
fwd connections: 1 in use, 2 most used
dir connections: 1 in use, 127 most used
centralized connections: 0 in use, 0 most used
VPN redirect connections: 0 in use, 0 most used
Inspect Snort:
preserve-connection: 0 enabled, 0 in effect, 1 most enabled, 0 most in effect

```

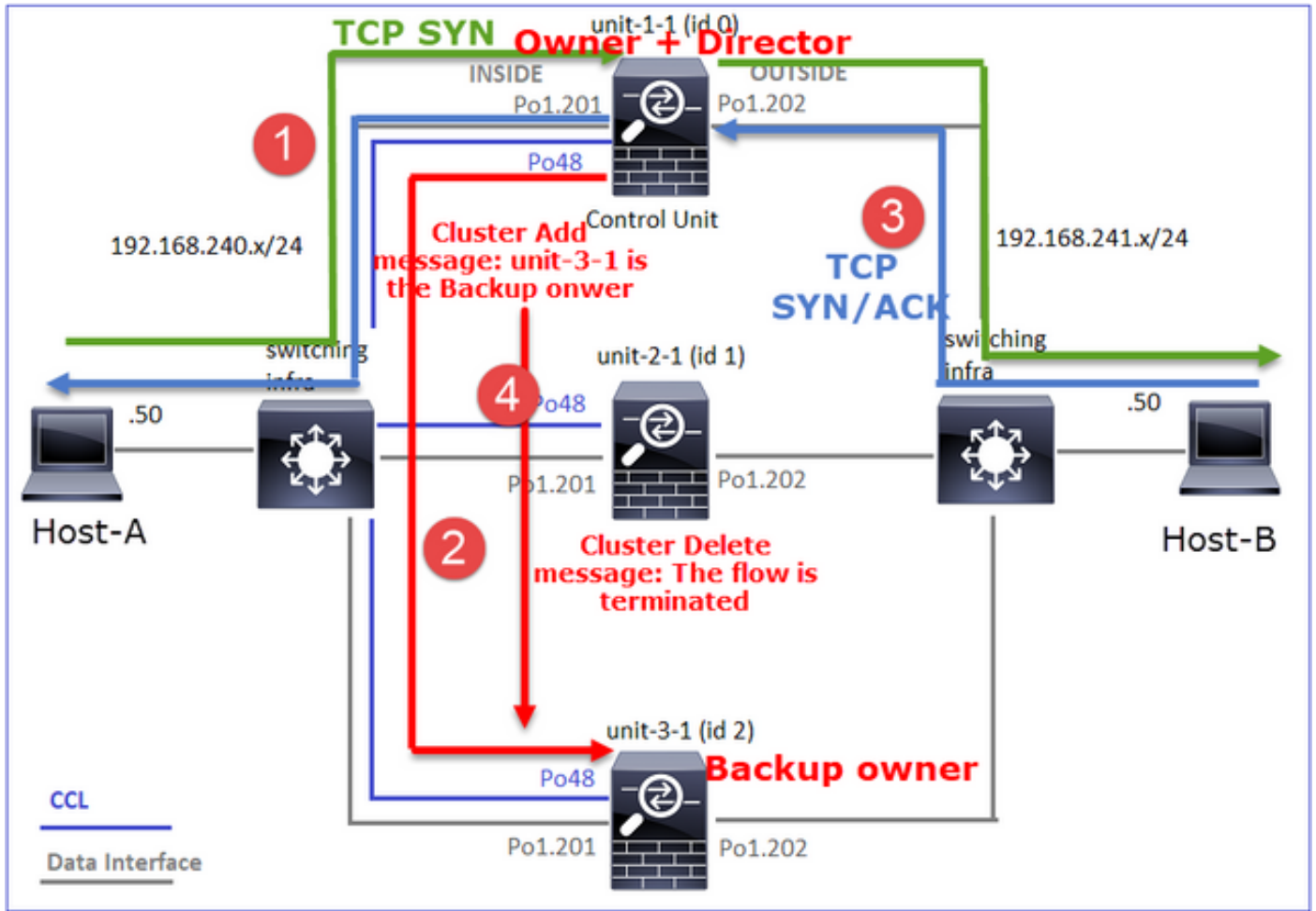
```

TCP OUTSIDE 192.168.241.50:443 NP Identity Ifc 192.168.240.50:39698, idle 0:00:23, bytes 0, flags z
TCP OUTSIDE 192.168.241.50:80 INSIDE 192.168.240.50:
45954
, idle 0:00:06, bytes 0,
flags y

```

單位	標誌	附註
Unit-1-1	UIO	·流所有者 — 裝置處理流 ·控制器 — 由於unit-3-1具有「y」而不是「Y」，這意味著已選擇unit-1-1作為此流的控制器。因此，由於它也是所有者，因此另一個單元（在本例中為unit-3-1）被選為備份所有者
Unit-2-1	-	-
Unit-3-1	y	裝置是備份所有者

其視覺化結果為：



1. TCP SYN資料包從主機A到達裝置1-1。裝置1-1成為流所有者。
2. Unit-1-1也被選為流導向器。因此，它還會選擇unit-3-1作為備份所有者（集群新增消息）。
3. TCP SYN/ACK封包從主機B到達裝置3-1。流量是對稱的。
4. 一旦連線終止，所有者將傳送群集刪除消息以從備份所有者中刪除流資訊。

觀察3.追蹤捕捉顯示兩個方向只通過1-1裝置。

步驟1.根據源埠，確定所有集群單元中關注的流和資料包：

```
<#root>
```

```
firepower#
```

```
cluster exec show capture CAPI | i 45954
```

```
unit-1-1(LOCAL):*****
```

```
1: 08:42:09.362697 802.1Q vlan#201 P0 192.168.240.50.45954 > 192.168.241.50.80: S 992089269:992089269(0
```

```
2: 08:42:09.363521 802.1Q vlan#201 P0 192.168.241.50.80 > 192.168.240.50.45954: S 4042762409:4042762409
```

```
3: 08:42:09.363827 802.1Q vlan#201 P0 192.168.240.50.45954 > 192.168.241.50.80: . ack 4042762410 win 22
```

```
...
```

```
unit-2-1:*****
```

```
unit-3-1:*****
```

```
<#root>
```

```
firepower#
```

```
cluster exec show capture CAPO | i 45954
```

```
unit-1-1(LOCAL):*****
```

```
1: 08:42:09.362987 802.1Q vlan#202 PO 192.168.240.50.45954 > 192.168.241.50.80: S 2732339016:2732339016
```

```
2: 08:42:09.363415 802.1Q vlan#202 PO 192.168.241.50.80 > 192.168.240.50.45954: S 3603655982:3603655982
```

```
3: 08:42:09.363903 802.1Q vlan#202 PO 192.168.240.50.45954 > 192.168.241.50.80: . ack 3603655983 win 22
```

```
...
```

```
unit-2-1:*****
```

```
unit-3-1:*****
```

步驟2.由於這是TCP流跟蹤3次握手資料包。從輸出中可看出，unit-1-1是所有者。為簡單起見，忽略不相關的跟蹤階段：

```
<#root>
```

```
firepower#
```

```
show cap CAPI packet-number 1 trace
```

```
25985 packets captured
```

```
1: 08:42:09.362697 802.1Q vlan#201 PO 192.168.240.50.
```

```
45954
```

```
> 192.168.241.50.80:
```

```
S
```

```
992089269:992089269(0) win 29200 <mss 1460,sackOK,timestamp 495153655 0,nop,wscale 7>
```

```
...
```

```
Phase: 4
```

```
Type: CLUSTER-EVENT
```

```
Subtype:
```

```
Result: ALLOW
```

```
Config:
```

```
Additional Information:
```

```
Input interface: 'INSIDE'
```

```
Flow type: NO FLOW
```

```
I (0) got initial, attempting ownership.
```


Phase: 5

Type: CLUSTER-EVENT

Subtype:

Result: ALLOW

Config:

Additional Information:

Input interface: 'INSIDE'

Flow type: NO FLOW

I (0) am becoming owner

...

返回流量(TCP SYN/ACK):

<#root>

firepower#

show capture CAPO packet-number 2 trace

25985 packets captured

2: 08:42:09.363415 802.1Q vlan#202 P0 192.168.241.50.80 > 192.168.240.50.45954:

S

3603655982:3603655982(0)

ack

2732339017 win 28960 <mss 1460,sackOK,timestamp 505509125 495153655,nop,wscale 7>

...

Phase: 3

Type: FLOW-LOOKUP

Subtype:

Result: ALLOW

Config:

Additional Information:

Found flow with id 9364, using existing flow

觀察4. FTD資料平面系統顯示所有裝置上的連線建立和終止：

<#root>

firepower#

```
cluster exec show log | include 45954
```

```
unit-1-1
```

```
(LOCAL):*****
```

```
Dec 01 2020 08:42:09: %FTD-6-302013:
```

```
Built inbound TCP connection 9364
```

```
for INSIDE:192.168.240.50/45954 (192.168.240.50/45954) to OUTSIDE:192.168.241.50/80 (192.168.241.50/80)
```

```
Dec 01 2020 08:42:18: %FTD-6-302014:
```

```
Teardown TCP connection 9364
```

```
for INSIDE:192.168.240.50/45954 to OUTSIDE:192.168.241.50/80 duration 0:00:08 bytes 1024000440 TCP FIN
```

```
unit-2-1:*****
```

```
unit-3-1
```

```
:*****
```

```
Dec 01 2020 08:42:09: %FTD-6-302022:
```

```
Built backup stub TCP connection
```

```
for INSIDE:192.168.240.50/45954 (192.168.240.50/45954) to OUTSIDE:192.168.241.50/80 (192.168.241.50/80)
```

```
Dec 01 2020 08:42:18: %FTD-6-302023:
```

```
Teardown backup TCP connection
```

```
for INSIDE:192.168.240.50/45954 to OUTSIDE:192.168.241.50/80 duration 0:00:08 forwarded bytes 0 Cluste
```

案例研究2.對稱流量 (所有者與控制器不同)

- 與案例研究相同#1但在本案例研究中，流量所有者與控制器是不同的單位。
- 所有輸出都與案例研究#1類似。與案例研究#1比較的主要差異是替代方案1的「y」標誌的「Y」標誌。

意見1.業主與所長不同。

源埠為46278的流的連線標誌分析。

```
<#root>
```

```
firepower#
```

```
cluster exec show conn
```

```
unit-1-1(LOCAL):*****
```

```
23 in use, 25 most used
```

```
Cluster:
```

```
fwd connections: 0 in use, 1 most used
```

```
dir connections: 0 in use, 122 most used
```

```
centralized connections: 0 in use, 0 most used
```

```
VPN redirect connections: 0 in use, 0 most used
```

```

Inspect Snort:
preserve-connection: 2 enabled, 0 in effect, 4 most enabled, 1 most in effect

TCP OUTSIDE 192.168.241.50:80 INSIDE 192.168.240.50:
46278
, idle 0:00:00, bytes 508848268, flags
UIO N1

TCP OUTSIDE 192.168.241.50:80 INSIDE 192.168.240.50:46276, idle 0:00:03, bytes 0, flags aA N1

unit-2-1:*****
21 in use, 271 most used
Cluster:
fwd connections: 0 in use, 2 most used
dir connections: 0 in use, 2 most used
centralized connections: 0 in use, 0 most used
VPN redirect connections: 0 in use, 0 most used
Inspect Snort:
preserve-connection: 0 enabled, 0 in effect, 249 most enabled, 0 most in effect

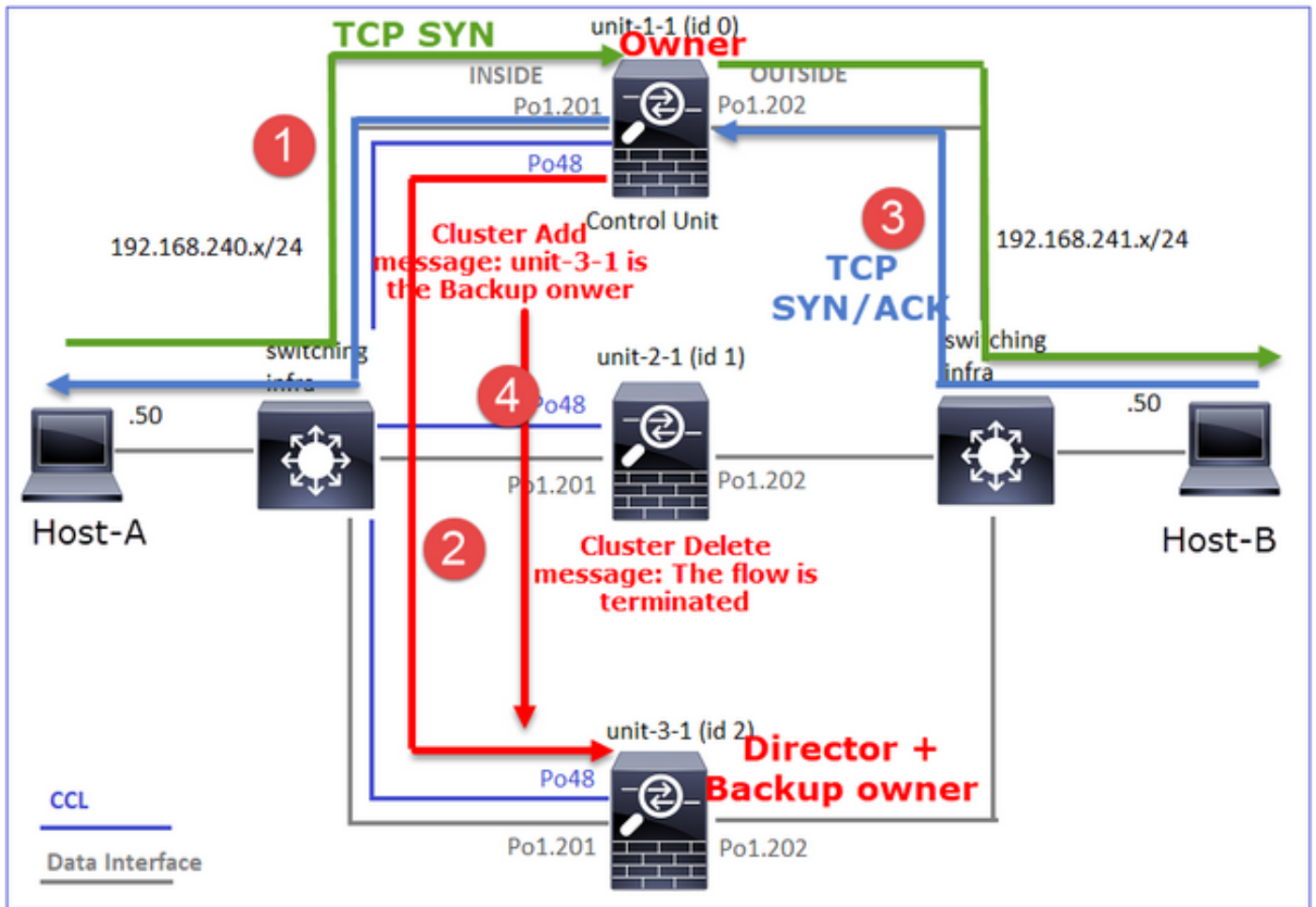
unit-3-1:*****
17 in use, 20 most used
Cluster:
fwd connections: 1 in use, 5 most used
dir connections: 1 in use, 127 most used
centralized connections: 0 in use, 0 most used
VPN redirect connections: 0 in use, 0 most used
Inspect Snort:
preserve-connection: 0 enabled, 0 in effect, 1 most enabled, 0 most in effect

TCP OUTSIDE 192.168.241.50:80 NP Identity Ifc 192.168.240.50:46276, idle 0:00:02, bytes 0, flags z
TCP OUTSIDE 192.168.241.50:80 INSIDE 192.168.240.50:
46278
, idle 0:00:06, bytes 0,
flags Y

```

單位	標誌	附註
Unit-1-1	UIO	·流所有者 — 裝置處理流
Unit-2-1	-	-
Unit-3-1	Y	·控制器和備份所有者 — 裝置3-1的標誌Y (控制器) 。

其視覺化結果為：



1. TCP SYN資料包從主機A到達裝置1-1。裝置1-1成為流所有者。
2. Unit-3-1被選為流導向器。Unit-3-1也是備份所有者 (UDP 4193上通過CCL傳送的「cluster add」消息)。
3. TCP SYN/ACK封包從主機B到達裝置3-1。流量是對稱的。
4. 連線終止後，所有者通過CCL在UDP 4193上傳送「群集刪除」消息，以從備份所有者中刪除流資訊。

觀察2.追蹤捕獲顯示兩個方向只通過1-1裝置

步驟1.使用與案例研究1相同的方法，根據源埠識別所有集群單元中的相關流和資料包：

```
<#root>
```

```
firepower#
```

```
cluster exec show cap CAPI | include 46278
```

```
unit-1-1
```

```
(LOCAL):*****
```

```
3: 11:01:44.841631 802.1Q vlan#201 P0 192.168.240.50.46278 > 192.168.241.50.80:
```

```
s
```

```
1972783998:1972783998(0) win 29200 <mss 1460,sackOK,timestamp 503529072 0,nop,wscale 7>
```

4: 11:01:44.842317 802.1Q vlan#201 PO 192.168.241.50.80 > 192.168.240.50.46278:

s

3524167695:3524167695(0)

ack

1972783999 win 28960 <mss 1380,sackOK,timestamp 513884542 503529072,nop,wscale 7>

5: 11:01:44.842592 802.1Q vlan#201 PO 192.168.240.50.46278 > 192.168.241.50.80: . ack 3524167696 win 22

unit-2-1:*****

unit-3-1:*****
firepower#

在OUTSIDE介面上擷取：

<#root>

firepower#

cluster exec show cap CAPO | include 46278

unit-1-1

(LOCAL):*****

3: 11:01:44.841921 802.1Q vlan#202 PO 192.168.240.50.46278 > 192.168.241.50.80:

s

2153055699:2153055699(0) win 29200 <mss 1380,sackOK,timestamp 503529072 0,nop,wscale 7>

4: 11:01:44.842226 802.1Q vlan#202 PO 192.168.241.50.80 > 192.168.240.50.46278:

s

3382481337:3382481337(0)

ack

2153055700 win 28960 <mss 1460,sackOK,timestamp 513884542 503529072,nop,wscale 7>

5: 11:01:44.842638 802.1Q vlan#202 PO 192.168.240.50.46278 > 192.168.241.50.80: . ack 3382481338 win 22

unit-2-1:*****

unit-3-1:*****
firepower#

步驟2.集中處理輸入封包 (TCP SYN和TCP SYN/ACK)：

<#root>

firepower#

```
cluster exec show cap CAPI packet-number 3 trace
```

```
unit-1-1(LOCAL):*****
```

```
824 packets captured
```

```
3: 11:01:44.841631 802.1Q vlan#201 P0 192.168.240.50.46278 > 192.168.241.50.80:
```

```
S
```

```
1972783998:1972783998(0) win 29200 <mss 1460,sackOK,timestamp 503529072 0,nop,wscale 7>
```

```
...
```

```
Phase: 4
```

```
Type: CLUSTER-EVENT
```

```
Subtype:
```

```
Result: ALLOW
```

```
Config:
```

```
Additional Information:
```

```
Input interface: 'INSIDE'
```

```
Flow type: NO FLOW
```

```
I (0) got initial, attempting ownership.
```

```
Phase: 5
```

```
Type: CLUSTER-EVENT
```

```
Subtype:
```

```
Result: ALLOW
```

```
Config:
```

```
Additional Information:
```

```
Input interface: 'INSIDE'
```

```
Flow type: NO FLOW
```

```
I (0) am becoming owner
```

在unit-1-1上跟蹤SYN/ACK:

```
<#root>
```

```
firepower#
```

```
cluster exec show cap CAPO packet-number 4 trace
```

```
unit-1-1(LOCAL):*****
```

```
4: 11:01:44.842226 802.1Q vlan#202 P0 192.168.241.50.80 > 192.168.240.50.
```

```
46278
```

```
:
```

```
S
```

3382481337:3382481337(0)

ack

2153055700 win 28960 <mss 1460,sackOK,timestamp 513884542 503529072,nop,wscale 7>

Phase: 3

Type: FLOW-LOOKUP

Subtype:

Result: ALLOW

Config:

Additional Information:

Found flow with id 9583, using existing flow

觀察3. FTD資料平面系統會顯示所有者和備份所有者上的連線建立和終止：

<#root>

firepower#

cluster exec show log | include 46278

unit-1-1(LOCAL):*****

Dec 01 2020 11:01:44: %FTD-6-302013:

Built inbound TCP connection

9583 for INSIDE:192.168.240.50/46278 (192.168.240.50/46278) to OUTSIDE:192.168.241.50/80 (192.168.241.50/80)

Dec 01 2020 11:01:53: %FTD-6-302014:

Teardown TCP connection

9583 for INSIDE:192.168.240.50/46278 to OUTSIDE:192.168.241.50/80 duration 0:00:08 bytes 1024001808 TC

unit-2-1:*****

unit-3-1:*****

Dec 01 2020 11:01:44: %FTD-6-302022:

Built director stub TCP connection

for INSIDE:192.168.240.50/46278 (192.168.240.50/46278) to OUTSIDE:192.168.241.50/80 (192.168.241.50/80)

Dec 01 2020 11:01:53: %FTD-6-302023:

Teardown director TCP connection

for INSIDE:192.168.240.50/46278 to OUTSIDE:192.168.241.50/80 duration 0:00:08 forwarded bytes 0 Cluste

案例研究3.非對稱流量 (指揮交換機轉發流量)。

觀察1. reinject-hide捕獲顯示單元1-1和單元2-1 (非對稱流) 上的資料包：

<#root>

firepower#

cluster exec show cap

unit-1-1(LOCAL):*****
capture CCL type raw-data buffer 33554432 interface cluster [Buffer Full - 33554320 bytes]
capture CAPI type raw-data buffer 100000 trace interface INSIDE [Buffer Full - 98552 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
capture CAPO type raw-data buffer 100000 trace interface OUTSIDE [Buffer Full - 98552 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
capture CAPI_RH type raw-data

reinject-hide

buffer 100000 interface

INSIDE

[Buffer Full -

98552 bytes

]

match tcp host 192.168.240.50 host 192.168.241.50 eq www

capture CAPI_RH type raw-data

reinject-hide

buffer 100000 interface

OUTSIDE

[Buffer Full -

99932 bytes

]

match tcp host 192.168.240.50 host 192.168.241.50 eq www

unit-2-1:*****

capture CCL type raw-data buffer 33554432 interface cluster [Buffer Full - 33553268 bytes]

capture CAPI type raw-data buffer 100000 trace interface INSIDE [Capturing - 0 bytes]

match tcp host 192.168.240.50 host 192.168.241.50 eq www

capture CAPO type raw-data buffer 100000 trace interface OUTSIDE [Buffer Full - 99052 bytes]

match tcp host 192.168.240.50 host 192.168.241.50 eq www

capture CAPI_RH type raw-data reinject-hide buffer 100000 interface INSIDE [Capturing - 0 bytes]

match tcp host 192.168.240.50 host 192.168.241.50 eq www

capture CAPO_RH type raw-data

reinject-hide

buffer 100000 interface

OUTSIDE

[Buffer Full -

99052 bytes

]

match tcp host 192.168.240.50 host 192.168.241.50 eq www

unit-3-1:*****

capture CCL type raw-data buffer 33554432 interface cluster [Capturing - 53815 bytes]

capture CAPI type raw-data buffer 100000 trace interface INSIDE [Capturing - 0 bytes]

match tcp host 192.168.240.50 host 192.168.241.50 eq www

capture CAPO type raw-data buffer 100000 trace interface OUTSIDE [Capturing - 658 bytes]

match tcp host 192.168.240.50 host 192.168.241.50 eq www


```
capture CAPI_RH type raw-data reinject-hide buffer 100000 interface INSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
capture CAPO_RH type raw-data reinject-hide buffer 100000 interface OUTSIDE [Capturing - 658 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
```

觀察2.源埠為46502的流的連線標誌分析。

```
<#root>
```

```
firepower#
```

```
cluster exec show conn
```

```
unit-1-1
```

```
(LOCAL):*****
```

```
23 in use, 25 most used
```

```
Cluster:
```

```
fwd connections: 0 in use, 1 most used
```

```
dir connections: 0 in use, 122 most used
```

```
centralized connections: 0 in use, 0 most used
```

```
VPN redirect connections: 0 in use, 0 most used
```

```
Inspect Snort:
```

```
preserve-connection: 2 enabled, 0 in effect, 4 most enabled, 1 most in effect
```

```
TCP OUTSIDE 192.168.241.50:80 INSIDE 192.168.240.50:
```

```
46502
```

```
, idle 0:00:00, bytes 448760236,
```

```
flags UIO N1
```

```
TCP OUTSIDE 192.168.241.50:80 INSIDE 192.168.240.50:46500, idle 0:00:06, bytes 0, flags aA N1
```

```
unit-2-1
```

```
:*****
```

```
21 in use, 271 most used
```

```
Cluster:
```

```
fwd connections: 0 in use, 2 most used
```

```
dir connections: 1 in use, 2 most used
```

```
centralized connections: 0 in use, 0 most used
```

```
VPN redirect connections: 0 in use, 0 most used
```

```
Inspect Snort:
```

```
preserve-connection: 0 enabled, 0 in effect, 249 most enabled, 0 most in effect
```

```
TCP OUTSIDE 192.168.241.50:80 INSIDE 192.168.240.50:
```

```
46502
```

```
, idle 0:00:00, bytes 0,
```

```
flags Y
```

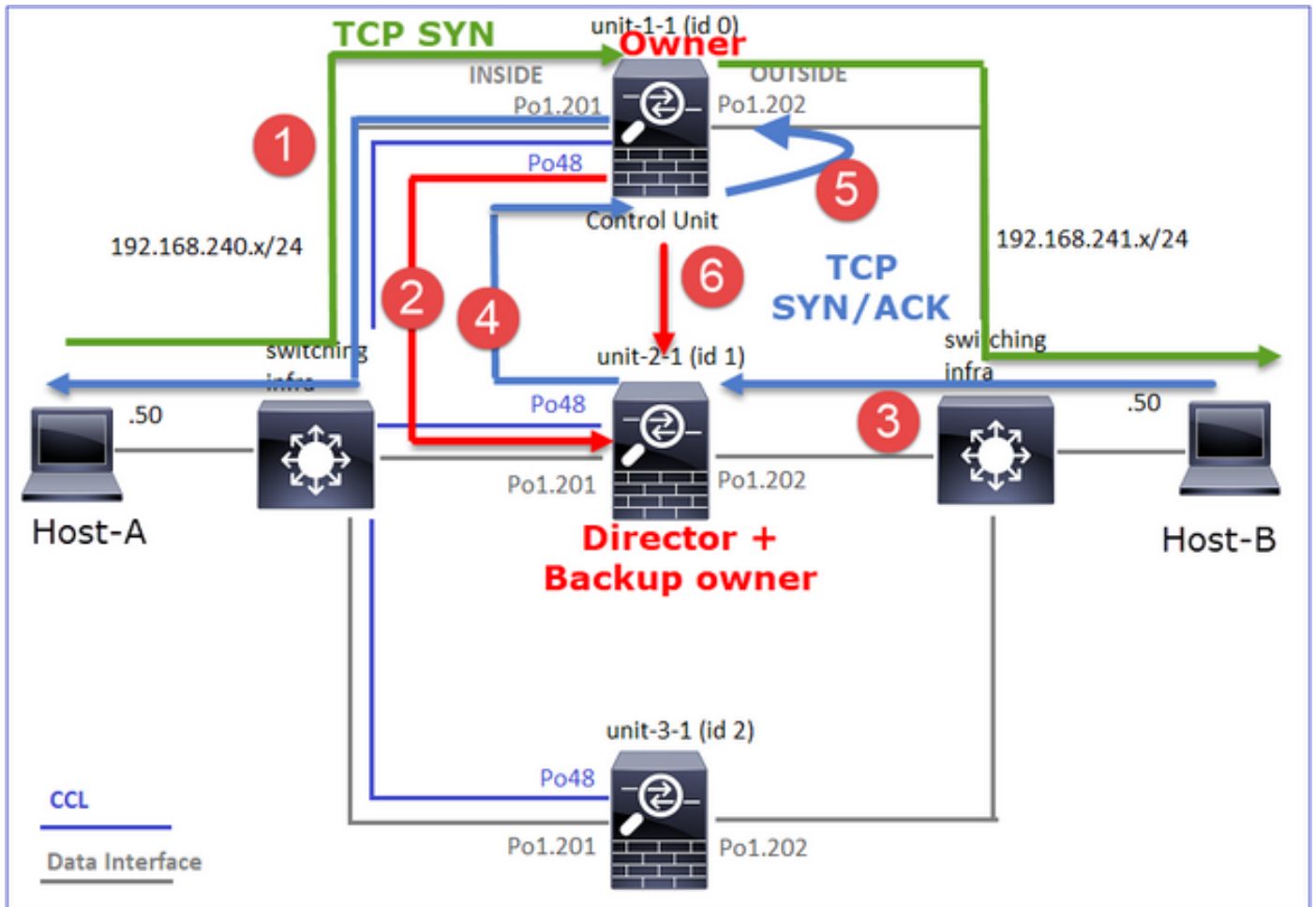
```

unit-3-1:*****
17 in use, 20 most used
Cluster:
fwd connections: 1 in use, 5 most used
dir connections: 0 in use, 127 most used
centralized connections: 0 in use, 0 most used
VPN redirect connections: 0 in use, 0 most used
Inspect Snort:
preserve-connection: 0 enabled, 0 in effect, 1 most enabled, 0 most in effect

```

單位	標誌	附註
Unit-1-1	UIO	·流所有者 — 裝置處理流。
Unit-2-1	Y	<p>·導向器 — 由於unit-2-1具有「Y」標籤，這意味著已選擇unit-2-1作為此流的導向器。</p> <p>·備份所有者</p> <p>·最後，雖然從該輸出中並不明顯，但是從show capture和show log輸出中，很明顯的unit-2-1會將此流轉發給所有者（儘管在本場景中技術上它並不被認為是轉發者）。</p> <p>附註：一個單位不能同時是導向器（Y流）和轉發器（z流），這兩個角色是互斥的。控制器（Y流）仍可以轉發流量。請參閱本案例分析後面的show log輸出。</p>
Unit-3-1	-	-

其視覺化結果為：



1. TCP SYN資料包從主機A到達裝置1-1。裝置1-1成為流所有者。
2. Unit-2-1被選為流導向器和備份所有者。流所有者在UDP 4193上傳送「cluster add」單播消息以通知備份所有者有關流的消息。
3. TCP SYN/ACK封包從主機B到達裝置2-1。流量是非對稱的。
4. Unit-2-1通過CCL將資料包轉發給所有者 (由於TCP SYN Cookie)。
5. 所有者重新在介面OUTSIDE上注入資料包，然後將資料包轉發到主機A。
6. 一旦連線終止，所有者將傳送群集刪除消息以從備份所有者中刪除流資訊。

觀察3.使用追蹤軌跡捕獲顯示非對稱流量以及從單元2-1重定向到單元1-1的過程。

步驟1.識別屬於關注流(連線埠46502)的封包：

```
<#root>
```

```
firepower#
```

```
cluster exec show capture CAPI | include 46502
```

```
unit-1-1(LOCAL):*****
```

```
3: 12:58:33.356121 802.1Q vlan#201 PO 192.168.240.50.46502 > 192.168.241.50.80: S 4124514680:4124514680
```

```
4: 12:58:33.357037 802.1Q vlan#201 PO 192.168.241.50.80 > 192.168.240.50.46502: S 883000451:883000451(0
```

```
5: 12:58:33.357357 802.1Q vlan#201 PO 192.168.240.50.46502 > 192.168.241.50.80: . ack 883000452 win 229
```

```
unit-2-1:*****
```

```
unit-3-1:*****
```

返回方向：

```
<#root>
```

```
firepower#
```

```
cluster exec show capture CAPO | include 46502
```

```
unit-1-1(LOCAL):*****
```

```
3: 12:58:33.356426 802.1Q vlan#202 PO 192.168.240.50.46502 > 192.168.241.50.80: S 1434968587:1434968587
```

```
4: 12:58:33.356915 802.1Q vlan#202 PO 192.168.241.50.80 > 192.168.240.50.46502: S 4257314722:4257314722
```

```
5: 12:58:33.357403 802.1Q vlan#202 PO 192.168.240.50.46502 > 192.168.241.50.80: . ack 4257314723 win 22
```

```
unit-2-1:*****
```

```
1: 12:58:33.359249 802.1Q vlan#202 PO 192.168.241.50.80 > 192.168.240.50.46502: S 4257314722:4257314722
```

```
2: 12:58:33.360302 802.1Q vlan#202 PO 192.168.241.50.80 > 192.168.240.50.46502: . ack 1434968736 win 23
```

```
3: 12:58:33.361004 802.1Q vlan#202 PO 192.168.241.50.80 > 192.168.240.50.46502: . 4257314723:4257316091
```

```
...
```

```
unit-3-1:*****
```

步驟2.追蹤封包。預設情況下，僅追蹤前50個輸入封包。為簡單起見，省略非相關的微量相。

Unit-1-1 (所有者)：

```
<#root>
```

```
firepower#
```

```
cluster exec show capture CAPI packet-number 3 trace
```

```
unit-1-1(LOCAL):*****
```

```
3: 12:58:33.356121 802.1Q vlan#201 PO 192.168.240.50.
```

```
46502
```

```
> 192.168.241.50.80:
```

```
S
```

```
4124514680:4124514680(0) win 29200 <mss 1460,sackOK,timestamp 510537534 0,nop,wscale 7>
```

```
...
```

```
Phase: 4
```

```
Type: CLUSTER-EVENT
```

```
Subtype:
```

```
Result: ALLOW
```

```
Config:
```

```
Additional Information:
```

```
Input interface: 'INSIDE'
```

```
Flow type: NO FLOW
```

```
I (0) got initial, attempting ownership.
```

Phase: 5
Type: CLUSTER-EVENT
Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'INSIDE'
Flow type: NO FLOW

I (0) am becoming owner

Unit-2-1 (轉發器)

返回流量(TCP SYN/ACK)。感興趣的單位是unit-2-1，該單位是控制器/備份的所有者並將流量轉發給所有者：

<#root>

firepower#

```
cluster exec unit unit-2-1 show capture CAPO packet-number 1 trace
```

```
1: 12:58:33.359249 802.1Q vlan#202 P0 192.168.241.50.80 > 192.168.240.50.
```

46502

```
: S 4257314722:4257314722(0) ack 1434968588 win 28960 <mss 1460,sackOK,timestamp 520893004 510537534,no
```

...

Phase: 4
Type: CLUSTER-EVENT
Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'OUTSIDE'
Flow type: NO FLOW

I (1) got initial, attempting ownership.

Phase: 5
Type: CLUSTER-EVENT
Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'OUTSIDE'
Flow type: NO FLOW

I (1) am early redirecting to (0) due to matching action (-1).

觀察4. FTD資料平面系統顯示所有裝置上的連線建立和終止：

```
<#root>
```

```
firepower#
```

```
cluster exec show log | i 46502
```

```
unit-1-1(LOCAL):*****
```

```
Dec 01 2020 12:58:33: %FTD-6-302013:
```

```
B
```

```
uilt inbound TCP connection
```

```
9742 for INSIDE:192.168.240.50/46502 (192.168.240.50/46502) to OUTSIDE:192.168.241.50/80 (192.168.241.50/80)
```

```
Dec 01 2020 12:59:02: %FTD-6-302014:
```

```
Teardown TCP connection
```

```
9742 for INSIDE:192.168.240.50/46502 to OUTSIDE:192.168.241.50/80 duration 0:00:28 bytes 2048000440 TC
```

```
unit-2-1:*****
```

```
Dec 01 2020 12:58:33: %FTD-6-302022:
```

```
Built forwarder stub TCP connection
```

```
for OUTSIDE:192.168.241.50/80 (192.168.241.50/80) to unknown:192.168.240.50/46502 (192.168.240.50/46502)
```

```
Dec 01 2020 12:58:33: %FTD-6-302023:
```

```
Teardown forwarder TCP connection
```

```
for OUTSIDE:192.168.241.50/80 to unknown:192.168.240.50/46502 duration 0:00:00 forwarded bytes 0 Forwa
```

```
Dec 01 2020 12:58:33: %FTD-6-302022:
```

```
Built director stub TCP connection
```

```
for INSIDE:192.168.240.50/46502 (192.168.240.50/46502) to OUTSIDE:192.168.241.50/80 (192.168.241.50/80)
```

```
Dec 01 2020 12:59:02: %FTD-6-302023:
```

```
Teardown director TCP connection
```

```
for INSIDE:192.168.240.50/46502 to OUTSIDE:192.168.241.50/80 duration 0:00:28 forwarded bytes 20483163
```

```
unit-3-1:*****
```

```
firepower#
```

案例研究4.非對稱流量 (所有者為主管)

觀察1. reinject-hide捕獲顯示單元1-1和單元2-1 (非對稱流) 上的資料包：

```
<#root>
```

```
firepower#
```

```
cluster exec show cap
```

```
unit-1-1(LOCAL):*****
```

```
capture CCL type raw-data buffer 33554432 interface cluster [Buffer Full - 33554229 bytes]
capture CAPI type raw-data buffer 100000 trace interface INSIDE [Buffer Full - 98974 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
capture CAPO type raw-data buffer 100000 trace interface OUTSIDE [Buffer Full - 98974 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
capture CAPI_RH type raw-data
```

reinject-hide

buffer 100000 interface

INSIDE

[Buffer Full -

98974 bytes

]

match tcp host 192.168.240.50 host 192.168.241.50 eq www

capture CAPI_RH type raw-data

reinject-hide

buffer 100000 interface

OUTSIDE

[Buffer Full -

99924 bytes

]

match tcp host 192.168.240.50 host 192.168.241.50 eq www

unit-2-1:*****

capture CCL type raw-data buffer 33554432 interface cluster [Buffer Full - 33552925 bytes]

capture CAPI type raw-data buffer 100000 trace interface INSIDE [Capturing - 0 bytes]

match tcp host 192.168.240.50 host 192.168.241.50 eq www

capture CAPO type raw-data buffer 100000 trace interface OUTSIDE [Buffer Full - 99052 bytes]

match tcp host 192.168.240.50 host 192.168.241.50 eq www

capture CAPI_RH type raw-data reinject-hide buffer 100000 interface INSIDE [Capturing - 0 bytes]

match tcp host 192.168.240.50 host 192.168.241.50 eq www

capture CAPO_RH type raw-data

reinject-hide

buffer 100000 interface OUTSIDE [Buffer Full -

99052 bytes

]

match tcp host 192.168.240.50 host 192.168.241.50 eq www

unit-3-1:*****

capture CCL type raw-data buffer 33554432 interface cluster [Capturing - 227690 bytes]

capture CAPI type raw-data buffer 100000 trace interface INSIDE [Capturing - 0 bytes]

match tcp host 192.168.240.50 host 192.168.241.50 eq www

capture CAPO type raw-data buffer 100000 trace interface OUTSIDE [Capturing - 4754 bytes]

match tcp host 192.168.240.50 host 192.168.241.50 eq www

capture CAPI_RH type raw-data reinject-hide buffer 100000 interface INSIDE [Capturing - 0 bytes]

match tcp host 192.168.240.50 host 192.168.241.50 eq www

capture CAPO_RH type raw-data reinject-hide buffer 100000 interface OUTSIDE [Capturing - 0 bytes]

match tcp host 192.168.240.50 host 192.168.241.50 eq www

觀察2.源埠為46916的流的連線標誌分析。

<#root>

firepower#

cluster exec show conn

unit-1-1

(LOCAL):*****

23 in use, 25 most used

Cluster:

fwd connections: 0 in use, 1 most used

dir connections: 0 in use, 122 most used

centralized connections: 0 in use, 0 most used

VPN redirect connections: 0 in use, 0 most used

Inspect Snort:

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

TCP OUTSIDE 192.168.241.50:80 INSIDE 192.168.240.50:

46916

, idle 0:00:00, bytes 414682616,

flags UIO N1

unit-2-1

:*****

21 in use, 271 most used

Cluster:

fwd connections: 1 in use, 2 most used

dir connections: 0 in use, 2 most used

centralized connections: 0 in use, 0 most used

VPN redirect connections: 0 in use, 0 most used

Inspect Snort:

preserve-connection: 0 enabled, 0 in effect, 249 most enabled, 0 most in effect

TCP OUTSIDE 192.168.241.50:80 NP Identity Ifc 192.168.240.50:

46916

, idle 0:00:00, bytes 0,

flags z

unit-3-1

:*****

17 in use, 20 most used

Cluster:


```

fwd connections: 0 in use, 5 most used
dir connections: 1 in use, 127 most used
centralized connections: 0 in use, 0 most used
VPN redirect connections: 0 in use, 0 most used
Inspect Snort:
preserve-connection: 0 enabled, 0 in effect, 1 most enabled, 0 most in effect

```

```
TCP OUTSIDE 192.168.241.50:80 INSIDE 192.168.240.50:
```

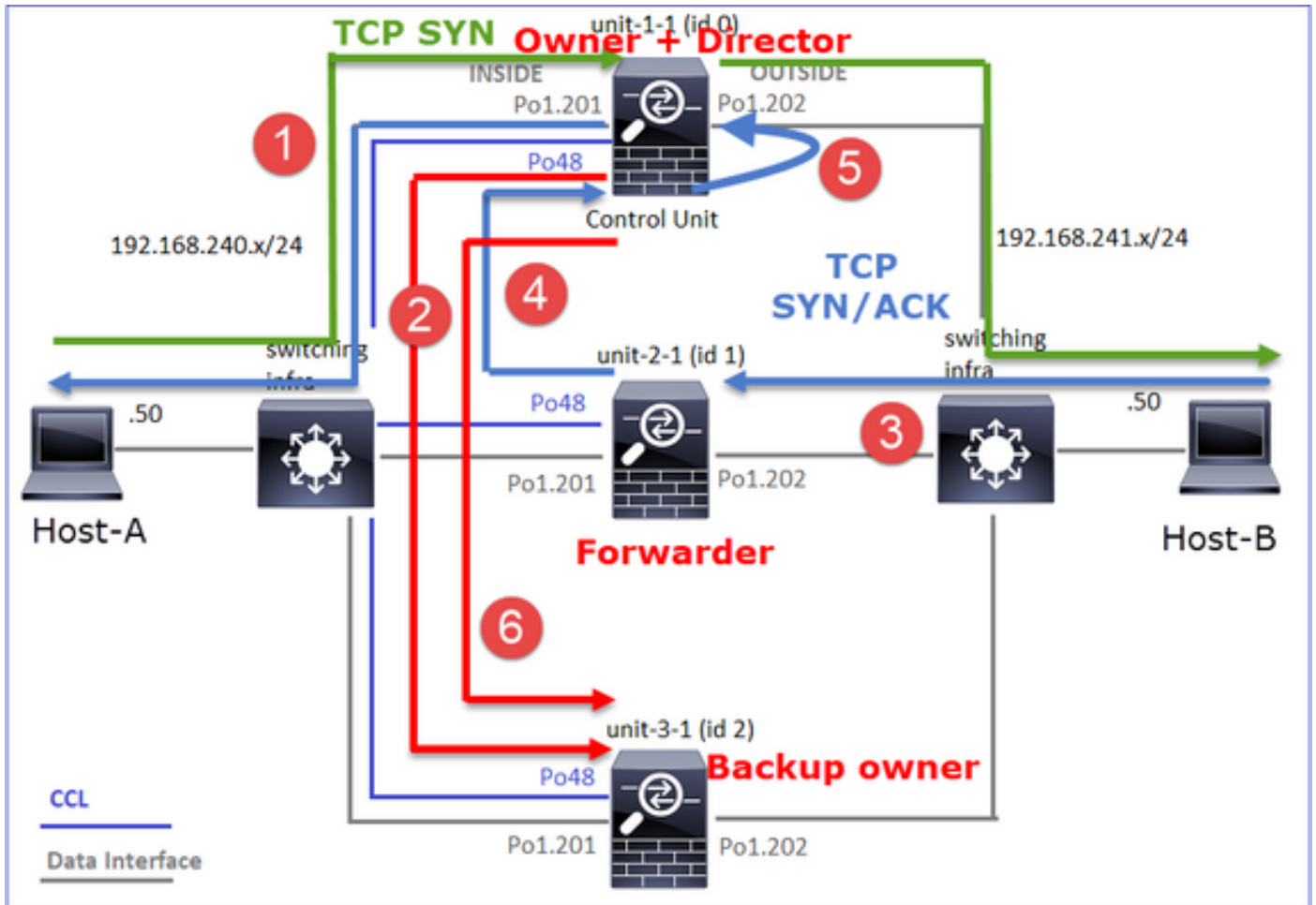
```
46916
```

```
, idle 0:00:04, bytes 0,
```

```
flags y
```

單位	標誌	附註
Unit-1-1	UIO	·流所有者 — 裝置處理流 ·控制器 — 由於unit-3-1具有「y」而不是「Y」，這意味著已選擇unit-1-1作為此流的控制器。因此，由於它也是所有者，因此另一個單元（在本例中為unit-3-1）被選為備份所有者
Unit-2-1	z	·轉發器
Unit-3-1	y	— 備份所有者

其視覺化結果為：



1. TCP SYN資料包從主機A到達裝置1-1。裝置1-1成為流的所有者，並被選為控制器。
2. Unit-3-1被選為備份所有者。流所有者在UDP 4193上傳送單播「cluster add」消息以通知備份所有者有關流的消息。
3. TCP SYN/ACK封包從主機B到達裝置2-1。流量是非對稱的。
4. Unit-2-1通過CCL將資料包轉發給所有者（由於TCP SYN Cookie）。
5. 所有者重新在介面OUTSIDE上注入資料包，然後將資料包轉發到主機A。
6. 一旦連線終止，所有者將傳送群集刪除消息以從備份所有者中刪除流資訊。

觀察3.使用追蹤軌跡捕獲顯示非對稱流量以及從單元2-1重定向到單元1-1的過程。

Unit-2-1 (轉發器)

<#root>

firepower#

cluster exec unit unit-2-1 show capture CAPO packet-number 1 trace

1: 16:11:33.653164 802.1Q vlan#202 PO 192.168.241.50.80 > 192.168.240.50.

46916

:

s

1331019196:1331019196(0)

ack

3089755618 win 28960 <mss 1460,sackOK,timestamp 532473211 522117741,nop,wscale 7>

...

Phase: 4

Type: CLUSTER-EVENT

Subtype:

Result: ALLOW

Config:

Additional Information:

Input interface: 'OUTSIDE'

Flow type: NO FLOW

I (1) got initial, attempting ownership.

Phase: 5

Type: CLUSTER-EVENT

Subtype:

Result: ALLOW

Config:

Additional Information:

Input interface: 'OUTSIDE'

Flow type: NO FLOW

I (1) am early redirecting to (0) due to matching action (-1).

觀察4. FTD資料平面系統顯示所有裝置上的連線建立和終止：

- Unit-1-1 (所有者)
- Unit-2-1 (轉發器)
- Unit-3-1 (備份所有者)

<#root>

firepower#

cluster exec show log | i 46916

unit-1-1(LOCAL):*****

Dec 01 2020 16:11:33: %FTD-6-302013:

Built inbound TCP connection

10023 for INSIDE:192.168.240.50/46916 (192.168.240.50/46916) to OUTSIDE:192.168.241.50/80 (192.168.241.50/80)

Dec 01 2020 16:11:42: %FTD-6-302014:

Teardown TCP connection

10023 for INSIDE:192.168.240.50/46916 to OUTSIDE:192.168.241.50/80 duration 0:00:09 bytes 1024010016 T

unit-2-1:*****

Dec 01 2020 16:11:33: %FTD-6-302022:

Built forwarder stub TCP connection

```
for OUTSIDE:192.168.241.50/80 (192.168.241.50/80) to unknown:192.168.240.50/46916 (192.168.240.50/46916)
Dec 01 2020 16:11:42: %FTD-6-302023:
```

```
Teardown forwarder TCP connection
```

```
for OUTSIDE:192.168.241.50/80 to unknown:192.168.240.50/46916 duration 0:00:09 forwarded bytes 1024009
```

```
unit-3-1:*****
Dec 01 2020 16:11:33: %FTD-6-302022:
```

```
Built backup stub TCP connection
```

```
for INSIDE:192.168.240.50/46916 (192.168.240.50/46916) to OUTSIDE:192.168.241.50/80 (192.168.241.50/80)
Dec 01 2020 16:11:42: %FTD-6-302023:
```

```
Teardown backup TCP connection
```

```
for INSIDE:192.168.240.50/46916 to OUTSIDE:192.168.241.50/80 duration 0:00:09 forwarded bytes 0 Cluste
```

案例研究5.非對稱流量 (所有者與控制器不同)。

觀察1. reinject-hide捕獲顯示單元1-1和單元2-1 (非對稱流) 上的資料包：

```
<#root>
```

```
firepower#
```

```
cluster exec show cap
```

```
unit-1-1
```

```
(LOCAL):*****
```

```
capture CCL type raw-data buffer 33554432 interface cluster [Buffer Full - 33553207 bytes]
capture CAPI type raw-data buffer 100000 trace interface INSIDE [Buffer Full - 99396 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
capture CAPO type raw-data buffer 100000 trace interface OUTSIDE [Buffer Full - 99224 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
capture CAPI_RH type raw-data
```

```
reinject-hide
```

```
buffer 100000 interface
```

```
INSIDE
```

```
[Buffer Full -
```

```
99396 bytes
```

```
]
```

```
match tcp host 192.168.240.50 host 192.168.241.50 eq www
capture CAPO_RH type raw-data
```

```
reinject-hid
```

```
e buffer 100000 interface
```

```
OUTSIDE
```

```
[Buffer Full -
```

99928 bytes

```
]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
```

unit-2-1

```
:*****
capture CCL type raw-data buffer 33554432 interface cluster [Buffer Full - 33554251 bytes]
capture CAPI type raw-data buffer 100000 trace interface INSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
capture CAPO type raw-data buffer 100000 trace interface OUTSIDE [Buffer Full - 99052 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
capture CAPI_RH type raw-data reinject-hide buffer 100000 interface INSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
capture CAPO_RH type raw-data
```

reinject-hide

buffer 100000 interface

OUTSIDE

[Buffer Full -

99052 bytes

```
]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
```

```
unit-3-1:*****
capture CCL type raw-data buffer 33554432 interface cluster [Capturing - 131925 bytes]
capture CAPI type raw-data buffer 100000 trace interface INSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
capture CAPO type raw-data buffer 100000 trace interface OUTSIDE [Capturing - 2592 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
capture CAPI_RH type raw-data reinject-hide buffer 100000 interface INSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
capture CAPO_RH type raw-data reinject-hide buffer 100000 interface OUTSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.241.50 eq www
```

觀察2.源埠為46994的流的連線標誌分析:

<#root>

firepower#

cluster exec show conn

unit-1-1

```
(LOCAL):*****
23 in use, 25 most used
Cluster:
fwd connections: 0 in use, 1 most used
dir connections: 0 in use, 122 most used
centralized connections: 0 in use, 0 most used
```

VPN redirect connections: 0 in use, 0 most used
 Inspect Snort:
 preserve-connection: 1 enabled, 0 in effect, 4 most enabled, 1 most in effect

TCP OUTSIDE 192.168.241.50:80 INSIDE 192.168.240.50:
 46994
 , idle 0:00:00, bytes 406028640,
 flags UIO N1

unit-2-1

:*****
 22 in use, 271 most used
 Cluster:
 fwd connections: 1 in use, 2 most used
 dir connections: 0 in use, 2 most used
 centralized connections: 0 in use, 0 most used
 VPN redirect connections: 0 in use, 0 most used
 Inspect Snort:
 preserve-connection: 0 enabled, 0 in effect, 249 most enabled, 0 most in effect

TCP OUTSIDE 192.168.241.50:80 NP Identity Ifc 192.168.240.50:
 46994
 , idle 0:00:00, bytes 0,
 flags z

unit-3-1

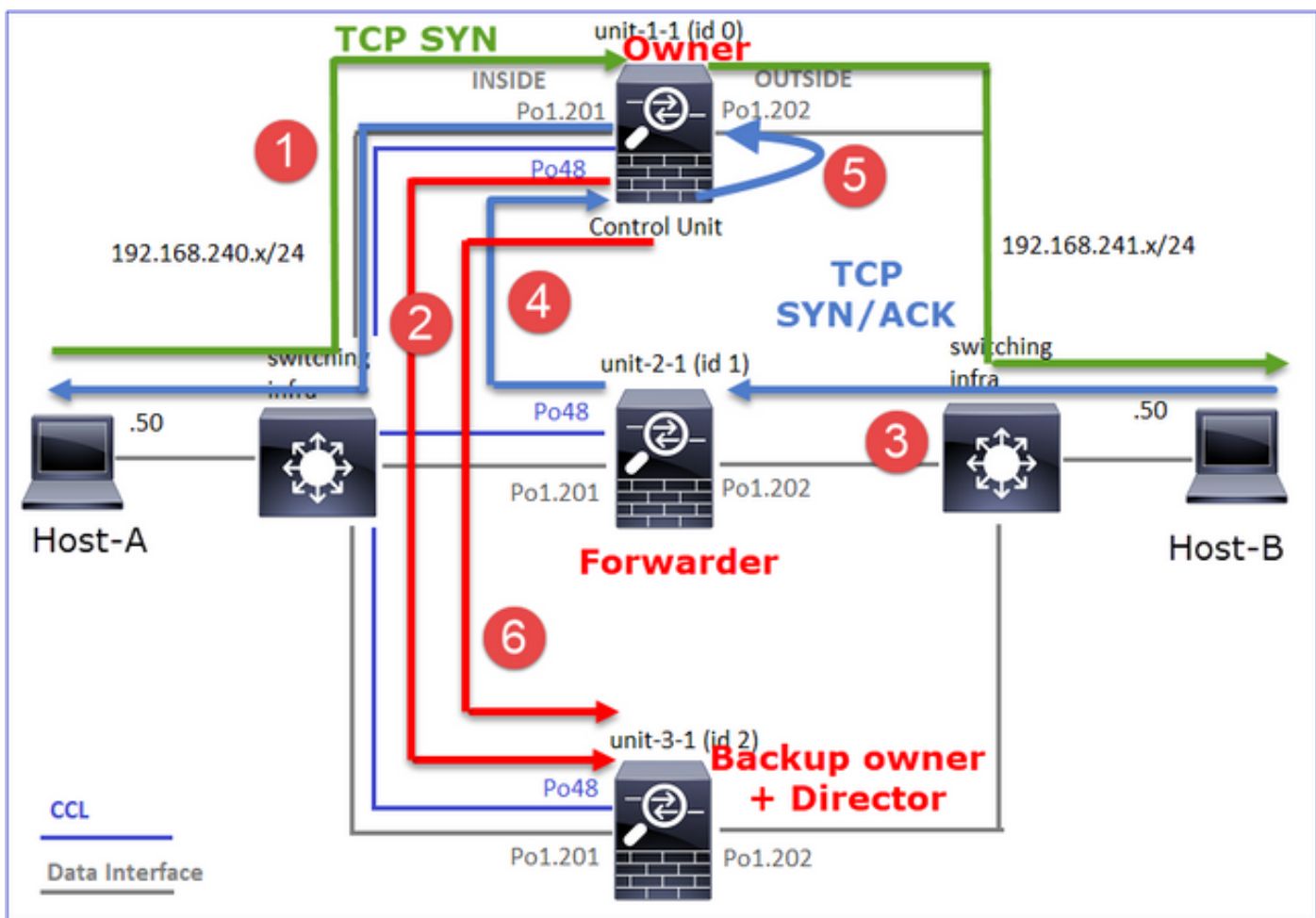
:*****
 17 in use, 20 most used
 Cluster:
 fwd connections: 2 in use, 5 most used
 dir connections: 1 in use, 127 most used
 centralized connections: 0 in use, 0 most used
 VPN redirect connections: 0 in use, 0 most used
 Inspect Snort:
 preserve-connection: 0 enabled, 0 in effect, 1 most enabled, 0 most in effect

TCP OUTSIDE 192.168.241.50:80 INSIDE 192.168.240.50:
 46994
 , idle 0:00:05, bytes 0,
 flags Y

單位	標誌	附註
----	----	----

Unit-1-1	UIO	·流所有者 — 裝置處理流
Unit-2-1	Z	·轉發器
Unit-3-1	Y	·備份所有者 ·董事

其視覺化結果為：



1. TCP SYN資料包從主機A到達裝置1-1。裝置1-1成為流所有者。
2. Unit-3-1被選為控制器和備份所有者。流所有者在UDP 4193上傳送「cluster add」單播消息以通知備份所有者有關流的消息。
3. TCP SYN/ACK封包從主機B到達裝置2-1。流量是非對稱的
4. Unit-2-1通過CCL將資料包轉發給所有者 (由於TCP SYN Cookie)。
5. 所有者重新在介面OUTSIDE上注入資料包，然後將資料包轉發到主機A。
6. 一旦連線終止，所有者將傳送群集刪除消息以從備份所有者中刪除流資訊。

觀察3.使用追蹤軌跡捕獲顯示非對稱流量以及從單元2-1重定向到單元1-1的過程。

Unit-1-1 (所有者)

<#root>

firepower#

cluster exec show cap CAPI packet-number 1 trace

unit-1-1(LOCAL):*****

...
Phase: 4
Type: CLUSTER-EVENT
Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'INSIDE'
Flow type: NO FLOW

I (0) got initial, attempting ownership.

Phase: 5
Type: CLUSTER-EVENT
Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'INSIDE'
Flow type: NO FLOW

I (0) am becoming owner

Unit-2-1 (轉發器)

<#root>

firepower#

cluster exec unit unit-2-1 show cap CAPO packet-number 1 trace

1: 16:46:44.232074 802.1Q vlan#202 P0 192.168.241.50.80 > 192.168.240.50.

46994

: S 2863659376:2863659376(0) ack 2879616990 win 28960 <mss 1460,sackOK,timestamp 534583774 524228304,no

...
Phase: 4
Type: CLUSTER-EVENT
Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'OUTSIDE'
Flow type: NO FLOW

I (1) got initial, attempting ownership.

Phase: 5
Type: CLUSTER-EVENT
Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'OUTSIDE'
Flow type: NO FLOW

I (1) am early redirecting to (0) due to matching action (-1).

觀察4. FTD資料平面系統顯示所有裝置上的連線建立和終止：

- Unit-1-1 (所有者)
- Unit-2-1 (轉發器)
- Unit-3-1 (備份所有者/控制器)

<#root>

firepower#

cluster exec show log | i 46994

unit-1-1(LOCAL):*****

Dec 01 2020 16:46:44: %FTD-6-302013:

Built inbound TCP connection

10080 for INSIDE:192.168.240.50/46994 (192.168.240.50/46994) to OUTSIDE:192.168.241.50/80 (192.168.241.50/80)
Dec 01 2020 16:46:53: %FTD-6-302014:

Teardown TCP connection

10080 for INSIDE:192.168.240.50/46994 to OUTSIDE:192.168.241.50/80 duration 0:00:09 bytes 1024000440 T

unit-2-1:*****

Dec 01 2020 16:46:44: %FTD-6-302022:

Built forwarder stub TCP connection

for OUTSIDE:192.168.241.50/80 (192.168.241.50/80) to unknown:192.168.240.50/46994 (192.168.240.50/46994)
Dec 01 2020 16:46:53: %FTD-6-302023:

Teardown forwarder TCP connection

for OUTSIDE:192.168.241.50/80 to unknown:192.168.240.50/46994 duration 0:00:09 forwarded bytes 1024000

unit-3-1:*****

Dec 01 2020 16:46:44: %FTD-6-302022:

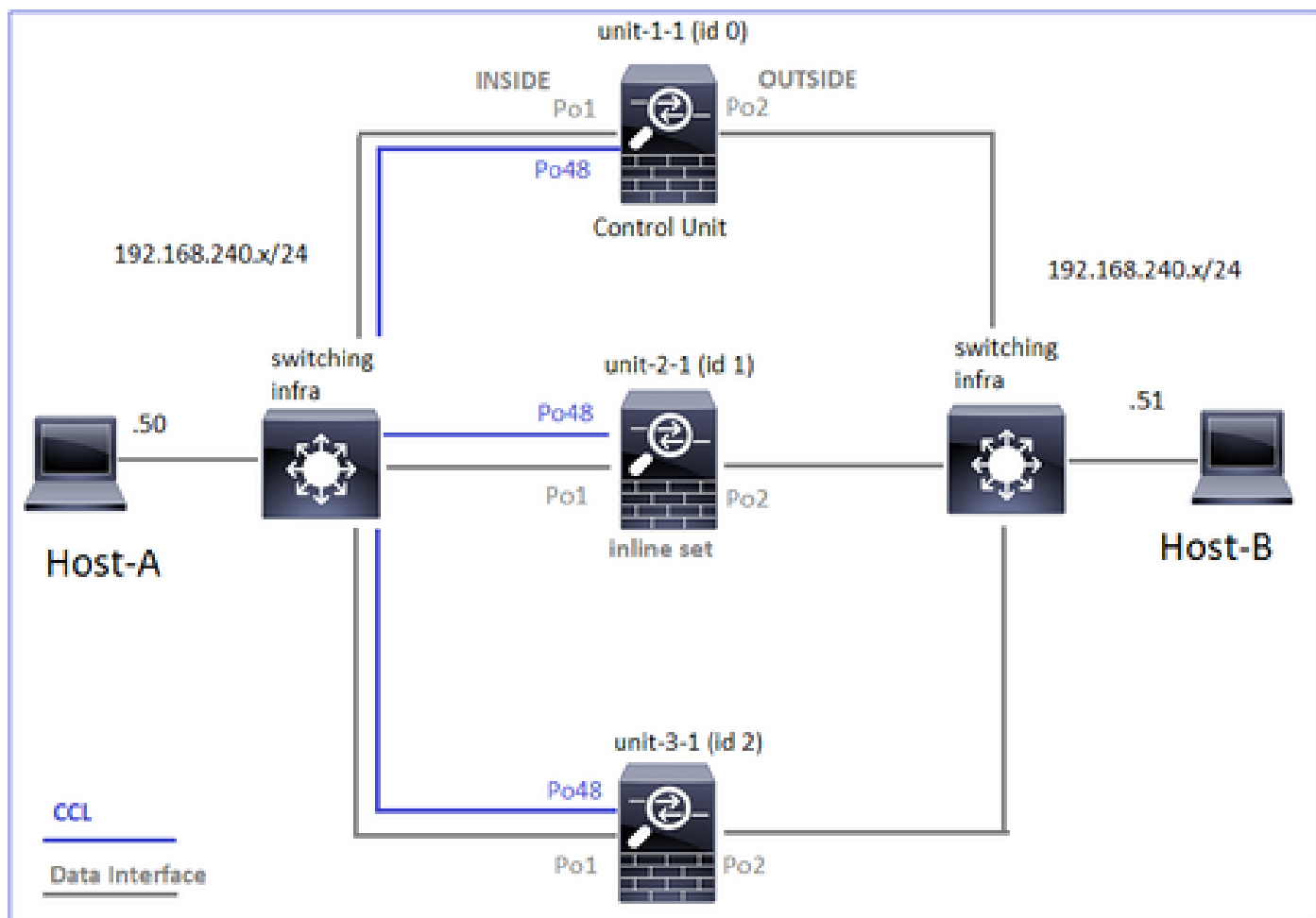
Built director stub TCP connection

for INSIDE:192.168.240.50/46994 (192.168.240.50/46994) to OUTSIDE:192.168.241.50/80 (192.168.241.50/80)
Dec 01 2020 16:46:53: %FTD-6-302023:

Teardown director TCP connection

for INSIDE:192.168.240.50/46994 to OUTSIDE:192.168.241.50/80 duration 0:00:09 forwarded bytes 0 Cluste

在下一個案例研究中，使用的拓撲基於具有內聯集的群集：



案例研究6.非對稱流量 (內嵌集，所有者為控制器)

觀察1. reinject-hide捕獲顯示單元1-1和單元2-1 (非對稱流) 上的資料包。此外，所有者是unit-2-1 (reinject-hide捕獲在INSIDE和OUTSIDE介面上都有資料包，而unit-1-1隻有OUTSIDE上的資料包)：

```
<#root>
```

```
firepower#
```

```
cluster exec show cap
```

```
unit-1-1
```

```
(LOCAL):*****  
capture CCL type raw-data buffer 33554432 interface cluster [Buffer Full - 33553253 bytes]  
capture CAPO type raw-data trace interface OUTSIDE [Buffer Full - 523432 bytes]  
match tcp host 192.168.240.50 host 192.168.240.51 eq www
```

```

capture CAPI type raw-data trace interface INSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.240.51 eq www
capture CAPO_RH type raw-data

reinject-hide

interface

OUTSIDE

[Buffer Full -

523432 bytes

]
match tcp host 192.168.240.50 host 192.168.240.51 eq www
capture CAPI_RH type raw-data reinject-hide interface INSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.240.51 eq www

unit-2-1

:*****
capture CCL type raw-data buffer 33554432 interface cluster [Buffer Full - 33554312 bytes]
capture CAPO type raw-data trace interface OUTSIDE [Buffer Full - 523782 bytes]
match tcp host 192.168.240.50 host 192.168.240.51 eq www
capture CAPI type raw-data trace interface INSIDE [Buffer Full - 523782 bytes]
match tcp host 192.168.240.50 host 192.168.240.51 eq www
capture CAPO_RH type raw-data

reinject-hide

interface

OUTSIDE

[Buffer Full -

524218 bytes

]
match tcp host 192.168.240.50 host 192.168.240.51 eq www
capture CAPI_RH type raw-data

reinject-hide

interface

INSIDE

[Buffer Full -

523782 bytes]

match tcp host 192.168.240.50 host 192.168.240.51 eq www

unit-3-1:*****
capture CCL type raw-data buffer 33554432 interface cluster [Capturing - 53118 bytes]
capture CAPO type raw-data trace interface OUTSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.240.51 eq www
capture CAPI type raw-data trace interface INSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.240.51 eq www
capture CAPO_RH type raw-data reinject-hide interface OUTSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.240.51 eq www
capture CAPI_RH type raw-data reinject-hide interface INSIDE [Capturing - 0 bytes]

```

```
match tcp host 192.168.240.50 host 192.168.240.51 eq www
```

觀察2.源埠為51844的流的連線標誌分析。

```
<#root>
```

```
firepower#
```

```
cluster exec show conn addr 192.168.240.51
```

```
unit-1-1
```

```
(LOCAL):*****
```

```
30 in use, 102 most used
```

```
Cluster:
```

```
fwd connections: 1 in use, 1 most used
```

```
dir connections: 2 in use, 122 most used
```

```
centralized connections: 3 in use, 39 most used
```

```
VPN redirect connections: 0 in use, 0 most used
```

```
Inspect Snort:
```

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

```
TCP OUTSIDE 192.168.240.51:80 NP Identity Ifc 192.168.240.50:
```

```
51844
```

```
, idle 0:00:00, bytes 0,
```

```
flags z
```

```
unit-2-1
```

```
:*****
```

```
23 in use, 271 most used
```

```
Cluster:
```

```
fwd connections: 0 in use, 2 most used
```

```
dir connections: 4 in use, 26 most used
```

```
centralized connections: 0 in use, 14 most used
```

```
VPN redirect connections: 0 in use, 0 most used
```

```
Inspect Snort:
```

```
preserve-connection: 0 enabled, 0 in effect, 249 most enabled, 0 most in effect
```

```
TCP OUTSIDE 192.168.240.51:80 INSIDE 192.168.240.50:
```

```
51844
```

```
, idle 0:00:00, bytes 231214400,
```

```
flags b N
```

```
unit-3-1
```

```
:*****
```

20 in use, 55 most used

Cluster:

fwd connections: 0 in use, 5 most used

dir connections: 1 in use, 127 most used

centralized connections: 0 in use, 24 most used

VPN redirect connections: 0 in use, 0 most used

Inspect Snort:

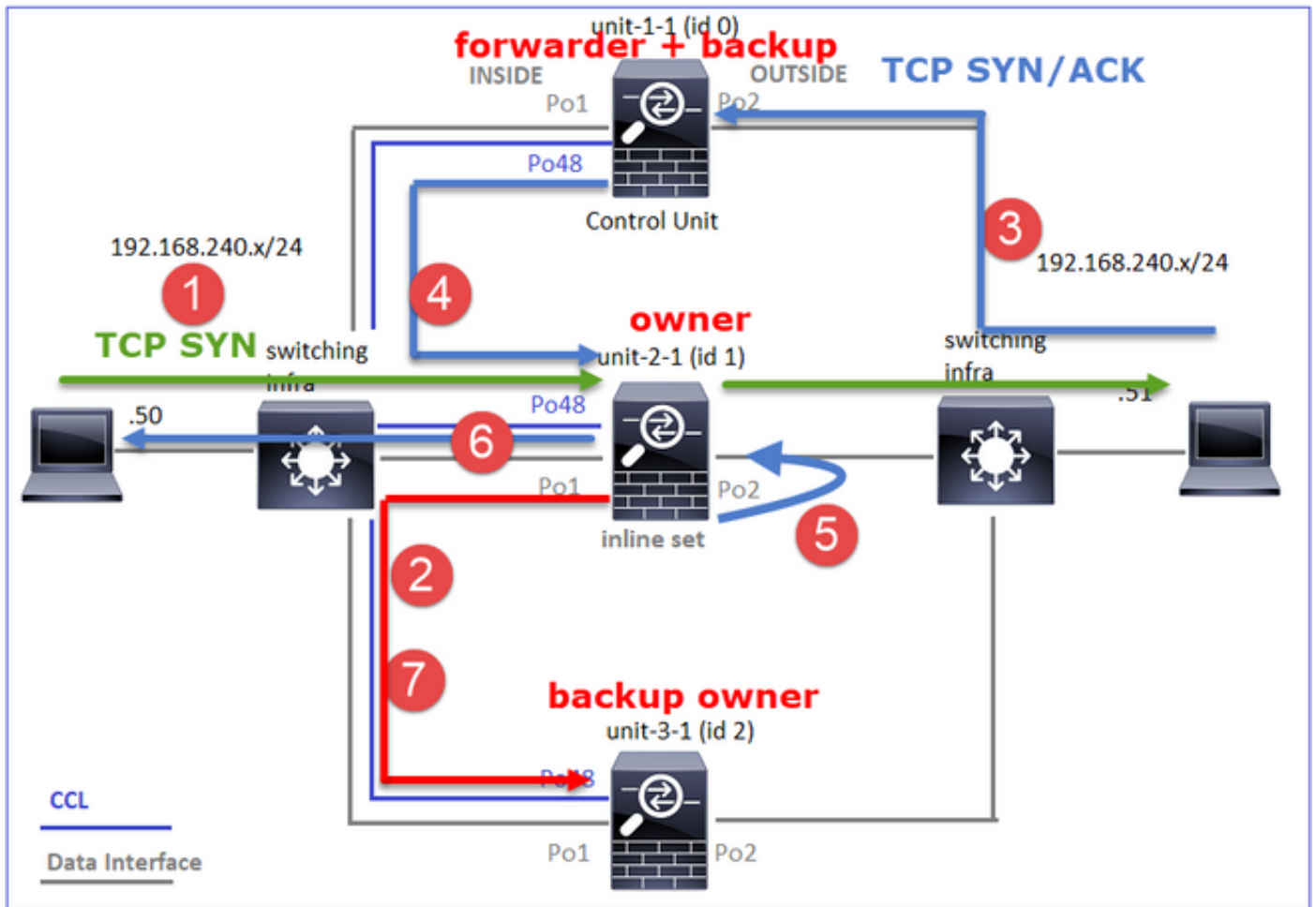
preserve-connection: 0 enabled, 0 in effect, 1 most enabled, 0 most in effect

TCP OUTSIDE 192.168.240.51:80 INSIDE 192.168.240.50:51844, idle 0:00:01, bytes 0,

flags y

單位	標誌	附註
Unit-1-1	z	·轉發器
Unit-2-1	b否	·流所有者 — 裝置處理流
Unit-3-1	y	·備份所有者

其視覺化結果為：



1. TCP SYN資料包從主機A到達裝置2-1。裝置2-1成為流所有者並被選為控制器。
2. Unit-3-1被選為備份所有者。流所有者在UDP 4193上傳送「cluster add」單播消息以通知備份所有者有關流的消息。
3. TCP SYN/ACK封包從主機B到達裝置1-1。流量是非對稱的。
4. Unit-1-1將資料包通過CCL轉發到控制器(unit-2-1)。
5. Unit-2-1也是所有者，它會在OUTSIDE介面上重新注入資料包。
6. Unit-2-1將資料包轉發到主機A。
7. 一旦連線終止，所有者將傳送群集刪除消息以從備份所有者中刪除流資訊。

觀察3.使用追蹤軌跡捕獲會顯示非對稱流量以及從unit-1-1重定向到unit-2-1的過程。

Unit-2-1 (所有者/主管)

```
<#root>
```

```
firepower#
```

```
cluster exec unit unit-2-1 show cap CAPI packet-number 1 trace
```

```
1: 18:10:12.842912 192.168.240.50.51844 > 192.168.240.51.80:
```

```
s
```

```
4082593463:4082593463(0) win 29200 <mss 1460,sackOK,timestamp 76258053 0,nop,wscale 7>
```

```
Phase: 1
```

```
Type: CLUSTER-EVENT
```

Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'INSIDE'
Flow type: NO FLOW

I (1) got initial, attempting ownership.

Phase: 2
Type: CLUSTER-EVENT
Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'INSIDE'
Flow type: NO FLOW

I (1) am becoming owner

Unit-1-1 (轉發器)

<#root>

firepower#

cluster exec show cap CAPO packet-number 1 trace

unit-1-1(LOCAL):*****

1: 18:10:12.842317 192.168.240.51.80 > 192.168.240.50.51844: S 2339579109:2339579109(0) ack 4082593464 v
Phase: 1
Type: CLUSTER-EVENT
Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'OUTSIDE'
Flow type: NO FLOW

I (0) am asking director (1).

返回流量(TCP SYN/ACK)

Unit-2-1 (所有者/主管)

<#root>

firepower#

```
cluster exec unit unit-2-1 show cap CAPO packet-number 2 trace
```

```
2: 18:10:12.843660 192.168.240.51.80 > 192.168.240.50.51844: S 2339579109:2339579109(0) ack 4082593464 v
Phase: 1
Type: CLUSTER-EVENT
Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'OUTSIDE'
Flow type: FULL
```

```
I (1) am owner, update sender (0).
```

```
Phase: 2
Type: FLOW-LOOKUP
Subtype:
Result: ALLOW
Config:
Additional Information:
```

```
Found flow with id 7109, using existing flow
```

觀察4. FTD資料平面系統顯示所有裝置上的連線建立和終止：

- Unit-1-1 (所有者)
- Unit-2-1 (轉發器)
- Unit-3-1 (備份所有者/控制器)

```
<#root>
```

```
firepower#
```

```
cluster exec show log | include 51844
```

```
unit-1-1(LOCAL):*****
```

```
Dec 02 2020 18:10:12: %FTD-6-302022:
```

```
Built forwarder stub TCP connection
```

```
for OUTSIDE:192.168.240.51/80 (192.168.240.51/80) to unknown:192.168.240.50/51844 (192.168.240.50/51844)
```

```
Dec 02 2020 18:10:22: %FTD-6-302023:
```

```
Teardown forwarder TCP connection
```

```
for OUTSIDE:192.168.240.51/80 to unknown:192.168.240.50/51844 duration 0:00:09 forwarded bytes 1024001
```

```
unit-2-1:*****
```

```
Dec 02 2020 18:10:12: %FTD-6-302303:
```

```
Built TCP state-bypass connection
```

```
7109 from INSIDE:192.168.240.50/51844 (192.168.240.50/51844) to OUTSIDE:192.168.240.51/80 (192.168.240.51/80)
```


Dec 02 2020 18:10:22: %FTD-6-302304:

Teardown TCP state-bypass connection

7109 from INSIDE:192.168.240.50/51844 to OUTSIDE:192.168.240.51/80 duration 0:00:09 bytes 1024001888 T

unit-3-1:*****

Dec 02 2020 18:10:12: %FTD-6-302022:

Built backup stub TCP connection

for INSIDE:192.168.240.50/51844 (192.168.240.50/51844) to OUTSIDE:192.168.240.51/80 (192.168.240.51/80)

Dec 02 2020 18:10:22: %FTD-6-302023:

Teardown backup TCP connection

for INSIDE:192.168.240.50/51844 to OUTSIDE:192.168.240.51/80 duration 0:00:09 forwarded bytes 0 Cluste

案例研究7.非對稱流量 (內嵌集 , 所有者與控制器不同)

所有者是unit-2-1 (reinject-hide捕獲在INSIDE和OUTSIDE介面上都有資料包 , 而unit-3-1隻有OUTSIDE上的資料包) :

<#root>

firepower#

cluster exec show cap

unit-1-1(LOCAL):*****

capture CCL type raw-data buffer 33554432 interface cluster [Capturing - 13902 bytes]

capture CAPO type raw-data trace interface OUTSIDE [Capturing - 90 bytes]

match tcp host 192.168.240.50 host 192.168.240.51 eq www

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

match tcp host 192.168.240.50 host 192.168.240.51 eq www

capture CAPO_RH type raw-data reinject-hide interface OUTSIDE [Capturing - 0 bytes]

match tcp host 192.168.240.50 host 192.168.240.51 eq www

capture CAPI_RH type raw-data reinject-hide interface INSIDE [Capturing - 0 bytes]

match tcp host 192.168.240.50 host 192.168.240.51 eq www

unit-2-1

:*****

capture CCL type raw-data buffer 33554432 interface cluster [Buffer Full - 33553936 bytes]

capture CAPO type raw-data trace interface OUTSIDE [Buffer Full - 523126 bytes]

match tcp host 192.168.240.50 host 192.168.240.51 eq www

capture CAPI type raw-data trace interface INSIDE [Buffer Full - 523126 bytes]

match tcp host 192.168.240.50 host 192.168.240.51 eq www

capture CAPO_RH type raw-data

reinject-hid

e

interface

OUTSIDE

[Buffer Full -

524230 bytes

```
]
match tcp host 192.168.240.50 host 192.168.240.51 eq www
capture CAPI_RH type raw-data
```

reinject-hide

interface

INSIDE

[Buffer Full -

523126 bytes

```
]
match tcp host 192.168.240.50 host 192.168.240.51 eq www
```

unit-3-1

```
:*****
capture CCL type raw-data buffer 33554432 interface cluster [Buffer Full - 33553566 bytes]
capture CAPO type raw-data trace interface OUTSIDE [Buffer Full - 523522 bytes]
match tcp host 192.168.240.50 host 192.168.240.51 eq www
capture CAPI type raw-data trace interface INSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.240.51 eq www
capture CAPO_RH type raw-data
```

reinject-hide

interface

OUTSIDE

[Buffer Full -

523432 bytes

```
]
match tcp host 192.168.240.50 host 192.168.240.51 eq www
capture CAPI_RH type raw-data reinject-hide interface INSIDE [Capturing - 0 bytes]
match tcp host 192.168.240.50 host 192.168.240.51 eq www
```

觀察2.源埠為59210的流的連線標誌分析。

<#root>

firepower#

cluster exec show conn addr 192.168.240.51

unit-1-1

```
(LOCAL):*****
25 in use, 102 most used
Cluster:
fwd connections: 0 in use, 1 most used
```

dir connections: 2 in use, 122 most used
centralized connections: 0 in use, 39 most used
VPN redirect connections: 0 in use, 0 most used
Inspect Snort:
preserve-connection: 0 enabled, 0 in effect, 4 most enabled, 1 most in effect

TCP OUTSIDE 192.168.240.51:80 INSIDE 192.168.240.50:

59210

, idle 0:00:03, bytes 0,

flags Y

unit-2-1

:*****

21 in use, 271 most used

Cluster:

fwd connections: 0 in use, 2 most used

dir connections: 0 in use, 28 most used

centralized connections: 0 in use, 14 most used

VPN redirect connections: 0 in use, 0 most used

Inspect Snort:

preserve-connection: 0 enabled, 0 in effect, 249 most enabled, 0 most in effect

TCP OUTSIDE 192.168.240.51:80 INSIDE 192.168.240.50:

59210

, idle 0:00:00, bytes 610132872,

flags b N

unit-3-1

:*****

19 in use, 55 most used

Cluster:

fwd connections: 1 in use, 5 most used

dir connections: 0 in use, 127 most used

centralized connections: 0 in use, 24 most used

VPN redirect connections: 0 in use, 0 most used

Inspect Snort:

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

TCP OUTSIDE 192.168.240.51:80 NP Identity Ifc 192.168.240.50:

59210

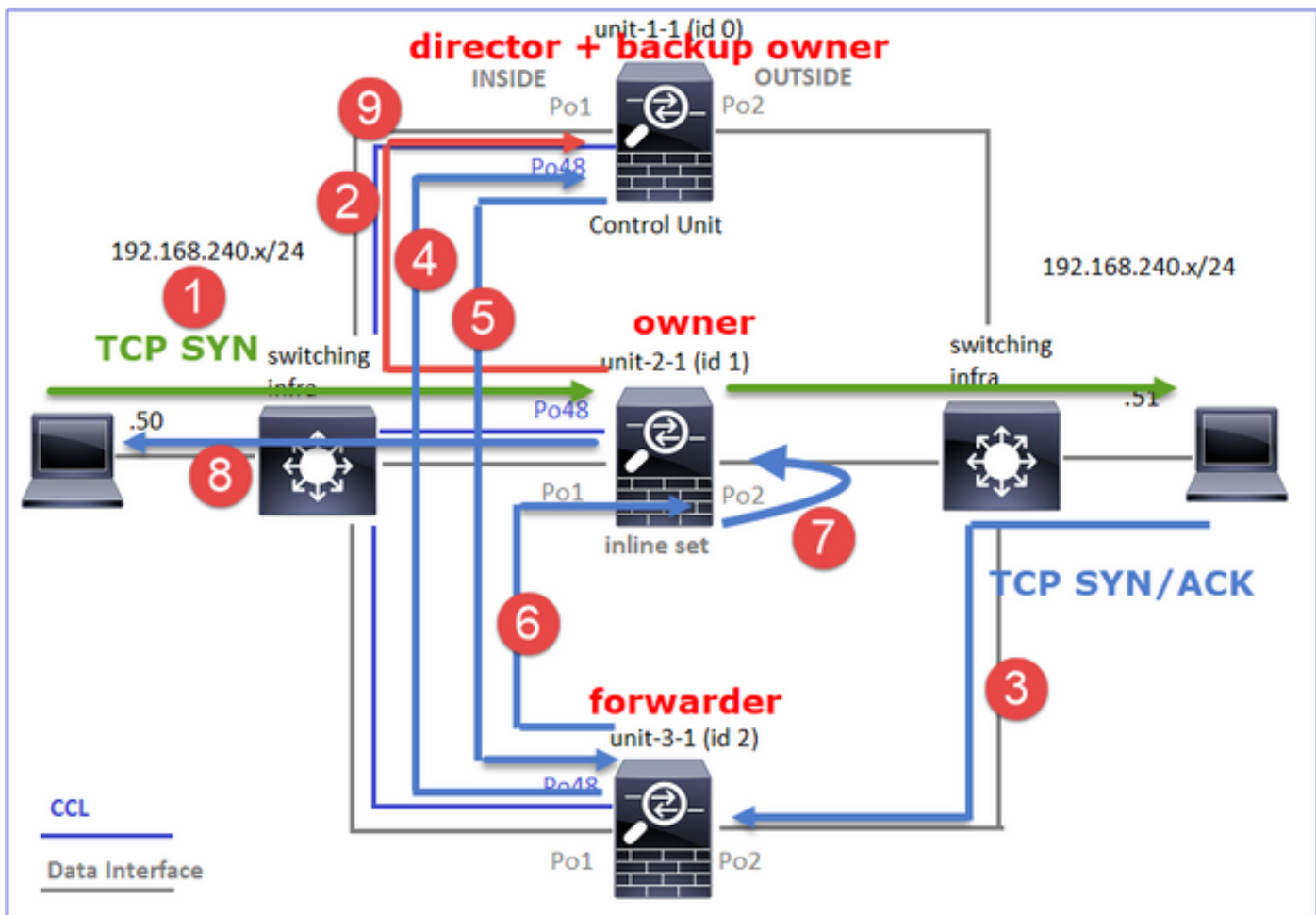
, idle 0:00:00, bytes 0,

flags z


單位	標誌	附註
----	----	----

Unit-1-1	Y	·控制器/備份所有者
Unit-2-1	b否	·流所有者 — 裝置處理流
Unit-3-1	z	·轉發器

其視覺化結果為：



1. TCP SYN資料包從主機A到達裝置2-1。裝置2-1成為流所有者，裝置1-1被選為控制器
2. Unit-1-1被選為備份所有者（因為它是主管）。流所有者將UDP 4193上的「cluster add」單播消息傳送到。通知備份所有者有關流量的資訊。
3. TCP SYN/ACK封包從主機B到達裝置3-1。流量是非對稱的。
4. Unit-3-1將資料包通過CCL轉發到控制器(unit-1-1)。
5. Unit-1-1(director)知道所有者是unit-2-1，將資料包傳送迴轉發器(unit-3-1)，並通知他所有者是unit-2-1。
6. Unit-3-1將資料包傳送到unit-2-1(owner)。
7. Unit-2-1在介面OUTSIDE上重新注入資料包。
8. Unit-2-1將資料包轉發到主機A。
9. 一旦連線終止，所有者將傳送群集刪除消息以從備份所有者中刪除流資訊。

 附註：步驟2 (封包通過CCL) 必須在步驟4 (資料流量) 之前執行。在其他情況下 (例如，競爭條件)，指揮交換機不知道流。因此，由於資料包是內嵌集，因此它會將資料包轉發到目的地。如果介面不在內嵌集內，資料封包會遭捨棄。

觀察3.使用追蹤軌跡捕獲會顯示非對稱流量和通過CCL的交換：

轉送流量(TCP SYN)

Unit-2-1 (所有者)

<#root>

firepower#

```
cluster exec unit unit-2-1 show cap CAPI packet-number 1 trace
```

```
1: 09:19:49.760702 192.168.240.50.59210 > 192.168.240.51.80: S 4110299695:4110299695(0) win 29200 <msg>
Phase: 1
Type: CLUSTER-EVENT
Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'INSIDE'
Flow type: NO FLOW
```

```
I (1) got initial, attempting ownership.
```

```
Phase: 2
Type: CLUSTER-EVENT
Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'INSIDE'
Flow type: NO FLOW
```

```
I (1) am becoming owner
```

返回流量(TCP SYN/ACK)

Unit-3-1 (ID 2 — 轉發器) 通過CCL將資料包傳送到unit-1-1(ID 0 - director)。

<#root>

firepower#

```
cluster exec unit unit-3-1 show cap CAPO packet-number 1 trace
```

1: 09:19:49.760336 192.168.240.51.80 > 192.168.240.50.59210:

s

4209225081:4209225081(0)

ack

4110299696 win 28960 <mss 1460,sackOK,timestamp 567715984 130834570,nop,wscale 7>
Phase: 1
Type: CLUSTER-EVENT
Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'OUTSIDE'
Flow type: NO FLOW

I (2) am asking director (0).

Unit-1-1(director)- Unit-1-1(ID 0)知道流所有者是unit-2-1(ID 1) , 並將資料包通過CCL傳送回unit-3-1 (ID 2 — 轉發器) 。

<#root>

firepower#

cluster exec show cap CAPO packet-number 1 trace

unit-1-1(LOCAL):*****

1: 09:19:49.761038 192.168.240.51.80 > 192.168.240.50.59210:

s

4209225081:4209225081(0)

ack

4110299696 win 28960 <mss 1460,sackOK,timestamp 567715984 130834570,nop,wscale 7>
Phase: 1
Type: CLUSTER-EVENT
Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'OUTSIDE'
Flow type: STUB

I (0) am director, valid owner (1), update sender (2).

Unit-3-1 (ID 2 — 轉發器) 通過CCL獲取資料包並將其傳送到unit-2-1 (ID 1 — 所有者) 。

<#root>

firepower#

```
cluster exec unit unit-3-1 show cap CAPO packet-number 2 trace
```

...

```
2: 09:19:49.761008 192.168.240.51.80 > 192.168.240.50.59210:
```

s

```
4209225081:4209225081(0) ack 4110299696 win 28960 <mss 1460,sackOK,timestamp 567715984 130834570,nop,w
```

Phase: 1

Type: CLUSTER-EVENT

Subtype:

Result: ALLOW

Config:

Additional Information:

Input interface: 'OUTSIDE'

Flow type: STUB

I (2) am becoming forwarder to (1), sender (0).

所有者重新將資料包轉發到目的地：

<#root>

firepower#

```
cluster exec unit unit-2-1 show cap CAPO packet-number 2 trace
```

```
2: 09:19:49.775701 192.168.240.51.80 > 192.168.240.50.59210:
```

s

```
4209225081:4209225081(0)
```

ack

```
4110299696 win 28960 <mss 1460,sackOK,timestamp 567715984 130834570,nop,wscale 7>
```

Phase: 1

Type: CLUSTER-EVENT

Subtype:

Result: ALLOW

Config:

Additional Information:

Input interface: 'OUTSIDE'

Flow type: FULL

I (1) am owner, sender (2).

觀察4. FTD資料平面系統顯示所有裝置上的連線建立和終止：

- Unit-1-1 (控制器/備份所有者)

- Unit-2-1 (所有者)
- Unit-3-1 (轉發器)

<#root>

firepower#

```
cluster exec show log | i 59210
```

```
unit-1-1(LOCAL):*****
```

```
Dec 03 2020 09:19:49: %FTD-6-302022:
```

```
Built director stub TCP connection
```

```
for INSIDE:192.168.240.50/59210 (192.168.240.50/59210) to OUTSIDE:192.168.240.51/80 (192.168.240.51/80)
```

```
Dec 03 2020 09:19:59: %FTD-6-302023:
```

```
Teardown director TCP connection
```

```
for INSIDE:192.168.240.50/59210 to OUTSIDE:192.168.240.51/80 duration 0:00:09 forwarded bytes 0 Cluste
```

```
unit-2-1:*****
```

```
Dec 03 2020 09:19:49: %FTD-6-302303:
```

```
Built TCP state-bypass connection
```

```
14483 from INSIDE:192.168.240.50/59210 (192.168.240.50/59210) to OUTSIDE:192.168.240.51/80 (192.168.240.51/80)
```

```
Dec 03 2020 09:19:59: %FTD-6-302304:
```

```
Teardown TCP state-bypass connection
```

```
14483 from INSIDE:192.168.240.50/59210 to OUTSIDE:192.168.240.51/80 duration 0:00:09 bytes 1024003336
```

```
unit-3-1:*****
```

```
Dec 03 2020 09:19:49: %FTD-6-302022:
```

```
Built forwarder stub TCP connection
```

```
for OUTSIDE:192.168.240.51/80 (192.168.240.51/80) to unknown:192.168.240.50/59210 (192.168.240.50/59210)
```

```
Dec 03 2020 09:19:59: %FTD-6-302023:
```

```
Teardown forwarder TCP connection
```

```
for OUTSIDE:192.168.240.51/80 to unknown:192.168.240.50/59210 duration 0:00:09 forwarded bytes 1024003
```

疑難排解

群集故障排除簡介

群集問題可分類為：

- 控制平面問題 (與群集穩定性相關的問題)
- 資料平面問題 (與傳輸流量相關的問題)

群集資料平面問題

NAT/PAT常見問題

重要配置注意事項

- 埠地址轉換(PAT)池的可用IP數必須至少與集群中的裝置數相等，最好是比集群節點多IP。
- 除非有特定原因禁用預設xlate per-session命令，否則必須保留這些命令。為禁用了xlate per-session的連線建立的任何PAT xlate始終由群集中的控制節點單元處理，這可能會導致效能降低。

高PAT池範圍使用率，因為源自低埠的流量會導致群集IP不平衡

FTD將PAT IP劃分為多個範圍，並嘗試將xlate保持在相同的來源範圍中。下表顯示如何將來源連線埠轉換為同一來源範圍內的全域連線埠。

原始Src埠	轉換後的Src埠
1-511	1-511
512-1023	512-1023
1024-65535	1024-65535

當來源連線埠範圍已滿且需要從該範圍分配新的PAT轉譯時，FTD會移動到下一個IP以為該來源連線埠範圍分配新的轉譯。

症狀

穿越集群的NAT流量的連線問題

驗證

```
<#root>
```

```
#
```

```
show nat pool
```

FTD資料平面記錄顯示PAT池耗盡：

```
<#root>
```

```
Dec 9 09:00:00 192.0.2.10 FTD-FW %ASA-3-202010:
```

```
PAT pool exhausted. Unable to create TCP connection
```

```
from Inside:192.0.2.150/49464 to Outside:192.0.2.250/20015
Dec 9 09:00:00 192.0.2.10 FTD-FW %ASA-3-202010:
PAT pool exhausted. Unable to create TCP connection
from Inside:192.0.2.148/54141 to Outside:192.0.2.251/443
```

緩解

配置NAT平面埠範圍並包括保留埠。

此外，在6.7/9.15.1之後，只有在節點離開/加入包含PAT的巨大後台流量的群集時，您才可能最終出現不均衡的埠塊分佈。它自己恢復的唯一方式是釋放埠塊以便在節點間重新分配。

使用基於埠塊的分發，當節點分配了大約10個埠塊（如pb-1、pb-2 ... pb-10）時。節點始終從第一個可用埠塊開始，並從該塊分配一個隨機埠，直到其耗盡。只有當到此時為止的所有埠塊都耗盡時，分配才會移動到下一個埠塊。

例如，如果主機建立了512個連線，則裝置會隨機為所有pb-1中的512個連線分配對映埠。現在，在所有512個連線都處於活動狀態時，當主機建立第513個連線時（因為pb-1已耗盡），它會移動到pb-2並從它分配一個隨機埠。現在，在513個連線中，假設第10個連線完成並清除了pb-1中的一個可用埠。此時，如果主機建立了第514個連線，集群單元將從pb-1而不是pb-2分配對映埠，因為pb-1現在有一個自由埠（在第10個連線刪除過程中釋放了該埠）。

需要記住的重要一點是，分配是從具有空閒埠的第一個可用埠塊開始的，這樣，在正常載入的系統中，最後一個埠塊始終可用於重分發。此外，PAT通常用於短期連線。埠塊在較短時間內變為可用的概率非常高。因此，使用基於埠塊的池分配，可以縮短池分配達到平衡所需的時間。

但是，如果從pb-1到pb-10的所有埠塊都已用盡，或者每個埠塊都有一個用於長期連線的埠，則這些埠塊永遠不會快速釋放並重新分配。在這種情況下，破壞性最小的方法是：

1. 識別埠塊過多的節點(show nat pool cluster summary)。
2. 確定該節點上最少使用的埠塊(show nat pool ip <addr> detail)。
3. 清除此類連線埠塊的xlates(clear xlate global <addr> gport 'start-end')，使其可用於重新分配。



警告：這會中斷相關連線。

當重定向到其他目標時，無法瀏覽到雙通道網站（如Web郵件、銀行等）或SSO網站。

症狀

無法瀏覽雙管道網站（如Web郵件、銀行網站等）。當使用者連線到要求客戶端開啟第二個套接字/連線的網站時，如果第二個連線被雜湊到與獲得第一個連線的群整合員不同的群整合員，並且流量使用IP PAT池，則當流量從其他公共IP地址接收連線時，伺服器會重置流量。

驗證

獲取資料平面群集捕獲以檢視如何處理受影響的傳輸流。在這種情況下，TCP重置來自目標網站。

緩解 (6.7/9.15.1之前的版本)

- 觀察是否有任何多會話應用程式使用多個對映的IP地址。
- 使用show nat pool cluster summary命令檢查池是否均勻分佈。
- 使用cluster exec show conn命令檢查流量是否正確負載均衡。
- 使用show nat pool cluster ip <address> detail命令檢查相粘IP的池使用情況。
- 啟用syslog305021錄(6.7/9.15)以檢視哪些連線未能使用粘性IP。
- 要解決向PAT池中新增更多IP或微調已連線交換機上的負載均衡演算法。

關於ether-channel負載平衡演算法：

- 對於非FP9300，如果身份驗證通過一台伺服器進行：調整相鄰交換機上從源IP/埠和目標IP/埠到源IP和目標IP的乙太網通道負載均衡演算法。
- 對於非FP9300，如果身份驗證通過多個伺服器進行：調整從來源IP/連線埠和目的地IP/連線埠到來源IP的相鄰交換器上的乙太網通道負載平衡演算法。
- 對於FP9300:在FP9300機箱上，負載均衡演算法固定為source-dest-port source-dest-ip source-dest-mac，並且無法更改。在此案例中，解決方法是使用FlexConfig將xlate per-session deny命令新增到FTD配置，強制某些目標IP地址（對於有問題/不相容的應用程式）的流量僅由機箱內群集中的控制節點處理。因應措施具有以下副作用：
 - 沒有以不同方式轉換的流量的負載均衡（所有內容都轉到控制節點）。
 - xlate插槽可能用盡（並對控制節點上其他流量的NAT轉換產生負面影響）。
 - 降低機箱內群集的可擴充性。

群集效能低，原因是池中的PAT IP不足，導致所有流量都傳送到控制節點。

症狀

集群中的PAT IP不足，無法向資料節點分配可用IP，因此，所有屬於PAT配置的流量都會轉發到控制節點進行處理。

驗證

使用show nat pool cluster命令檢視每台裝置的分配，並確認它們在池中至少擁有一個IP。

緩解

對於6.7/9.15.1之前的版本，請確保您的PAT池的大小至少等於群集中的節點數。在具有PAT池的6.7/9.15.1後版本中，可以從所有PAT池IP分配埠塊。如果PAT池使用率非常高，從而導致頻繁用盡池，則需要增加PAT池大小（請參見「常見問題」部分）。

由於未啟用每個會話，因此傳送到控制節點的所有流量都會導致效能較低。

症狀

通過集群控制節點處理大量高速UDP備份流，影響效能。

背景

只有使用xlate且已啟用每個會話的連線才能由使用PAT的資料節點處理。使用命令show run all xlate檢視xlate per-session config。

啟用每個會話意味著當關聯的連線斷開時，將立即關閉xlate。這有助於提高連線採用PAT時的每秒連線效能。在關聯連線斷開後，非每個會話的可用狀態將再延長30秒，如果連線速率足夠高，則每個全域性IP上可用的65k TCP/UDP埠可以在短時間內用完。

預設情況下，所有TCP流量都啟用每會話，只有UDP DNS流量啟用每會話。這表示所有非DNS UDP流量都轉發到控制節點進行處理。

驗證

使用以下命令檢查群集裝置之間的連線和資料包分佈：

```
<#root>
```

```
firepower#
```

```
show cluster info conn-distribution
```

```
firepower#
```

```
show cluster info packet-distribution
```

```
firepower#
```

```
show cluster info load-monitor
```

使用cluster exec show conn命令檢視哪些群集節點擁有UDP連線。

```
<#root>
```

```
firepower#
```

```
cluster exec show conn
```

使用此命令可以瞭解群集節點間的池使用情況。

```
<#root>
```

```
firepower#
```

```
cluster exec show nat pool ip
```

緩解

為相關流量（例如UDP）設定每個作業階段PAT(per-session permit udp 命令)。對於ICMP，您不能更改預設的多會話PAT，因此，當配置了PAT時，控制節點始終處理ICMP流量。

當節點離開/加入群集時，PAT池分佈變得不平衡。

症狀

- 連線問題，因為PAT IP分配可能會因裝置離開並加入集群而隨時間變得不平衡。
- 在6.7/9.15.1之後，新加入的節點可能無法獲得足夠的埠塊。沒有任何連線埠封鎖的節點會將流量重新導向到控制節點。至少具有一個埠塊的節點會處理流量，並在池耗盡後丟棄該流量。

驗證

- 資料平面系統日誌顯示如下消息：

```
<#root>
```

```
%ASA-3-202010:
```

```
NAT pool exhausted. Unable to create TCP connection  
from inside:192.0.2.1/2239 to outside:192.0.2.150/80
```

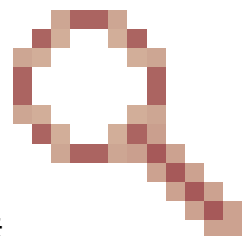
- 使用show nat pool cluster summary命令確定池分佈。
- 使用cluster exec show nat pool ip <addr> detail命令瞭解群集節點上的池使用情況。

緩解

- 10530對於6.7/9.15.1之前的版本，思科錯誤ID [CSCvd](#)中介紹了一些解決方法
- 在6.7/9.15.1之後，使用clear xlate global <ip> gport <start-end>命令手動清除其他節點上的某些埠塊，以便重新分配到所需節點。

症狀

集群通過PAT傳輸的流量的主要連線問題。這是因為FTD資料平面（根據設計）不傳送全域NAT位址的GARP。



驗證

直連裝置的ARP表顯示了更改控制節點後集群資料介面的MAC地址的不同：

```
<#root>
root@kali2:~/tests#
arp -a

? (192.168.240.1) at f4:db:e6:
33:44:2e

[ether] on eth0
root@kali2:~/tests#
arp -a

? (192.168.240.1) at f4:db:e6:
9e:3d:0e

[ether] on eth0
```

緩解

在群集資料介面上配置靜態（虛擬）MAC。

受PAT影響的連線失敗

症狀

集群通過PAT傳輸的流量的連線問題。

驗證/緩解

- 確保正確複製配置。
- 確保均勻分配池。
- 確保池所有權有效。
- show asp cluster counter中沒有失敗計數器增量。
- 確保使用正確資訊建立導向器/轉發器流。
- 驗證是否按預期建立、更新和清理了備份副本。
- 驗證是否根據「每個會話」行為建立和終止匯出。
- 啟用「debug nat 2」以指示任何錯誤。請注意，此輸出可能會非常嘈雜，例如：

```
<#root>
firepower#
debug nat 2
```

```
nat:
no free blocks available to reserve for 192.168.241.59, proto 17
```

```
nat: no free blocks available to reserve for 192.168.241.59, proto 17
nat: no free blocks available to reserve for 192.168.241.58, proto 17
nat: no free blocks available to reserve for 192.168.241.58, proto 17
nat: no free blocks available to reserve for 192.168.241.57, proto 17
```

要停止調試，請執行以下操作：

```
<#root>
firepower#
un all
```

- 啟用連線和NAT相關系統日誌，將資訊與故障連線相關聯。

ASA和FTD集群PAT改進 (9.15和6.7之後)

發生了什麼變化？

重新設計了PAT操作。單個IP不再分配給每個集群成員。相反，PAT IP被拆分為埠塊，並結合IP粘性操作在集群成員之間均勻 (儘可能) 分配這些埠塊。

新設計解決了這些限制 (請參見上一節)：

- 多會話應用程式會因缺乏群集範圍的IP粘性而受到影響。
- 要求的PAT池的大小至少等於群集中的節點數。
- 當節點離開/加入群集時，PAT池分佈變得不平衡。
- 沒有系統日誌指示PAT池不平衡。

從技術上講，PAT的預設埠範圍是1024-65535，而不是預設的1-511、512-1023和1024-65535埠範圍。此預設範圍可以擴展，以包括常規PAT的特權埠範圍1-1023 (「include-reserve」選項)。

這是FTD 6.7上的PAT池配置示例。有關其他詳細資訊，請檢視《配置指南》中的相關部分：

NAT Rule:
Manual NAT Rule

Insert:
In Category NAT Rules Before

Type:
Dynamic

Enable

Description:

Interface Objects Translation PAT Pool Advanced

Original Packet	Translated Packet
Original Source:* net_192.168.240.0	Translated Source: Address
Original Destination: Address	
Original Source Port:	Translated Source Port:
Original Destination Port:	Translated Destination Port:

Interface Objects Translation PAT Pool Advanced

Enable PAT Pool

PAT:
Address ip_192.168.241.57-59

Use Round Robin Allocation

Extended PAT Table

Flat Port Range ⓘ This option always enabled on device from v6.7.0 irrespective of its configured value.

Include Reserve Ports

Block Allocation

有關PAT的其他故障排除資訊

FTD資料平面系統日誌 (6.7/9.15.1之後)

當群集節點上的粘性IP中的所有埠都用完時，將生成一個粘性失效系統日誌，分配將移動到具有空

閒埠的下一個可用IP，例如：

```
%ASA-4-305021: Ports exhausted in pre-allocated PAT pool IP 192.0.2.100 for host 198.51.100.100 Allocat
```

當節點加入集群時，會在節點上生成池不平衡系統日誌，並且不會獲得埠塊的任何份額或不等份額，例如：

```
%ASA-4-305022: Cluster unit ASA-4 has been allocated 0 port blocks for PAT usage. All units should have  
%ASA-4-305022: Cluster unit ASA-4 has been allocated 12 port blocks for PAT usage. All units should have
```

Show命令

池分佈狀態

在show nat pool cluster summary輸出中，對於每個PAT IP地址，均衡分佈方案中的節點間差異不得超過1個埠塊。均衡和不均衡埠塊分佈的示例。

```
<#root>
```

```
firepower#
```

```
show nat pool cluster summary
```

```
port-blocks count display order: total, unit-1-1, unit-2-1, unit-3-1
```

```
IP OUTSIDE:ip_192.168.241.57-59 192.168.241.57 (126 -
```

```
42 / 42 / 42
```

```
)
```

```
IP OUTSIDE:ip_192.168.241.57-59 192.168.241.58 (126 - 42 / 42 / 42)
```

```
IP OUTSIDE:ip_192.168.241.57-59 192.168.241.59 (126 - 42 / 42 / 42)
```

分配不均衡：

```
<#root>
```

```
firepower#
```

```
show nat pool cluster summary
```

```
port-blocks count display order: total, unit-1-1, unit-4-1, unit-2-1, unit-3-1
```

```
IP outside:src_map 192.0.2.100 (128 - 32 /
```

```
22 / 38
```

```
/ 36)
```

池所有權狀態

在show nat pool cluster輸出中，不得存在所有者或備份為UNKNOWN的單個埠塊。如果存在，則表示池所有權通訊有問題。範例：

```
<#root>
```

```
firepower#
```

```
show nat pool cluster | in
```

```
[3072-3583], owner unit-4-1, backup <
```

```
UNKNOWN
```

```
>
```

```
[56832-57343], owner <UNKNOWN>, backup <UNKNOWN>
```

```
[10240-10751], owner unit-2-1, backup <UNKNOWN>
```

埠塊中埠分配的記帳

show nat pool命令通過其他選項得到增強，顯示詳細資訊和過濾後的輸出。範例：

```
<#root>
```

```
firepower#
```

```
show nat pool detail
```

```
TCP PAT pool INSIDE, address 192.168.240.1, range 1-1023, allocated 0
```

```
TCP PAT pool INSIDE, address 192.168.240.1, range 1024-65535, allocated 18
```

```
UDP PAT pool INSIDE, address 192.168.240.1, range 1-1023, allocated 0
```

```
UDP PAT pool INSIDE, address 192.168.240.1, range 1024-65535, allocated 20
```

```
TCP PAT pool OUTSIDE, address 192.168.241.1, range 1-1023, allocated 0
```

```
TCP PAT pool OUTSIDE, address 192.168.241.1, range 1024-65535, allocated 18
```

```
UDP PAT pool OUTSIDE, address 192.168.241.1, range 1-1023, allocated 0
```

```
UDP PAT pool OUTSIDE, address 192.168.241.1, range 1024-65535, allocated 20
```

```
UDP PAT pool OUTSIDE, address 192.168.241.58
```

```
range 1024-1535, allocated 512
```

```
range 1536-2047, allocated 512
```

```
range 2048-2559, allocated 512
```

```
range 2560-3071, allocated 512
```

```
...
```

```
unit-2-1:*****
```

```
UDP PAT pool OUTSIDE, address 192.168.241.57
```

```
range 1024-1535, allocated 512 *
```

```
range 1536-2047, allocated 512 *
```

```
range 2048-2559, allocated 512 *
```

「*」表示它是備份埠塊

要解決此問題，請使用clear xlate global <ip> gport <start-end>命令手動清除其他節點上的某些埠塊，以便重新分配到所需的節點。

手動觸發的埠塊重新分發

- 在流量恆定的生產網路中，當某個節點離開並重新加入群集時（可能是由於進行回溯），有時它無法獲得池的同等份額，或者在最糟糕的情況下，它無法獲得任何埠塊。
- 使用show nat pool cluster summary命令確定哪個節點擁有的埠塊多於所需數量。
- 在擁有更多埠塊的節點上，使用show nat pool ip <addr> detail命令找出分配數量最少的埠塊。
- 使用clear xlate global <address> gport <start-end>命令清除從這些埠塊建立的轉換，以便它們可以重新分發到所需的節點，例如：

```
<#root>
```

```
firepower#
```

```
show nat pool detail | i 19968
```

```
range 19968-20479, allocated 512
range 19968-20479, allocated 512
range 19968-20479, allocated 512
```

```
firepower#
```

```
clear xlate global 192.168.241.57 gport 19968-20479
```

```
INFO: 1074 xlates deleted
```

6.7/9.15.1之後PAT的常見問題(FAQ)

問：如果集群中可用單元數量的IP數量已足夠，是否仍可以將每單元1個IP用作選項？

答：現在不再如此，並且基於IP地址和基於埠塊的池分配方案之間沒有切換開關。

基於IP地址的池分發的舊方案導致多會話應用失敗，其中來自主機的多個連線（屬於單個應用事務的一部分）被負載均衡到群集的不同節點，從而被不同的對映IP地址轉換，導致目標伺服器看到它們來自不同的實體。

而且，使用基於埠塊的新分配方案，即使您現在可以使用低至單個PAT IP地址，也始終建議根據需要PAT的連線數量使用足夠的PAT IP地址。

問：是否仍可以為該群集的PAT池保留一個IP地址池？

是的，你可以。來自所有PAT池IP的埠塊分佈於群集節點。

問：如果對PAT池使用多個IP地址，則每個IP地址分配給每個成員的埠塊是否相同？

答：不，每個IP都是獨立分佈的。

問：所有群集節點都具有所有的公共IP，但只是部分埠？如果是這種情況，那麼能否保證每次源IP使用相同的公共IP？

A.正確，每個PAT IP由每個節點部分擁有。如果某個節點上的選定公有IP已用盡，則會生成系統日誌，指示不能保留粘性IP，並且分配將移動到下一個可用的公有IP。無論是獨立、HA或群集部署，IP粘性始終以盡力而為的方式進行，具體取決於池的可用性。

問：所有內容是否都基於PAT池中的單個IP地址，但是如果PAT池中使用了多個IP地址，則不適用？

A.它也適用於PAT池中的多個IP地址。來自PAT池中每個IP的埠塊分佈於群集節點。PAT池中的每個IP地址都會在群集中的所有成員之間拆分。因此，如果您在PAT池中有一個C類地址，則每個集群成員都有來自每個PAT池地址的埠池。

它和CGNAT公司合作嗎？

A.是的，也支援CGNAT。CGNAT（也稱為塊分配PAT）具有預設塊大小「512」，可以通過xlate塊分配大小CLI進行修改。在常規動態PAT（非CGNAT）的情況下，塊大小始終為「512」，這是固定且不可配置的。

問：如果裝置離開集群，控制節點是否將埠塊範圍分配給其他裝置或保留給自己？

A.每個埠塊都有一個所有者和備份。每次從埠塊建立xlate時，它也會複製到埠塊備份節點。當節點離開集群時，備份節點擁有所有埠塊和所有當前連線。備份節點由於已成為這些附加埠塊的所有者，因此會為其選擇新的備份，並將所有當前副本複製到該節點，以處理故障情況。

根據這個警報，我們可以採取什麼行動來增加粘性？

A.粘性無法保持有兩個可能的原因。

Reason-1:流量錯誤地進行了負載均衡，因為其中一個節點看到的連線數多於其他節點，導致特定的粘性IP耗盡。如果確保在群集節點上平均分配流量，則可以解決此問題。例如，在FPR41xx群集上，調整連線的交換機上的負載均衡演算法。在FPR9300群集上，確保機箱中有同等數量的刀片。

Reason-2: PAT池的使用率非常高，從而導致池的頻繁耗盡。要解決此問題，請增加PAT池大小。

問：如何處理對extended關鍵字之支援？它是否顯示錯誤，並阻止在升級期間新增整個NAT命令，還是刪除extended關鍵字並顯示警告？

A.從ASA 9.15.1/FP 6.7開始，集群不支援PAT擴展選項。配置選項不會從任何CLI/ASDM/CSM/FMC中刪除。配置（直接或間接通過升級）時，您將收到一條警告消息，該配置被接受，但您看不到正在運行的PAT的擴展功能。

問：是否與併發連線相同的轉換數？

A.在6.7/9.15.1之前的版本中，儘管它是1-65535，由於源埠在1-1024範圍內從未被大量使用，因此

它實際上是1024-65535(64512 conns)。在將「flat」作為預設行為的6.7/9.15.1後實現中，其值為1024-65535。但是，如果您希望使用1-1024，則可以使用「include-reserve」選項。

問：如果節點加入群集，它會將舊的備份節點用作備份，而那個備份節點會為其提供其舊的埠塊？

A.這取決於當時埠塊的可用性。當節點離開群集時，其所有埠塊都將移動到備份節點。然後是控制節點累積空閒埠塊並將其分發到所需節點。

問：如果控制節點的狀態發生更改，則選擇新的控制節點，是保留PAT塊分配，還是基於新的控制節點重新分配埠塊？

A.新控制節點瞭解已分配哪些塊，哪些塊是免費的，哪些是從那裡開始的。

問：此行為的最大併發連線數是否與最大併發連線數相同？

是。xlates的最大數量取決於PAT埠的可用性。這與最大併發連線數無關。如果僅允許1個地址，則可能有65535個連線。如果您需要更多，您必須分配更多IP地址。如果有足夠的地址/埠，您可以達到最大併發連線數。

問：新增新的集群成員時，埠塊分配過程是什麼？如果由於重新啟動而新增了集群成員，會發生什麼情況？

A.埠塊始終由控制節點分配。僅當存在可用埠塊時，才會將埠塊分配給新節點。自由埠塊表示不通過埠塊內的任何對映埠提供連線。

此外，在重新加入時，每個節點重新計算它可以擁有的塊數。如果節點擁有的塊數多於它應該擁有的塊數，它將在這些埠塊可用時將其釋放給控制節點。然後，控制節點將它們分配給新加入的資料節點。

問：它是否只支援TCP、UDP協定或SCTP？

A.動態PAT從未支援SCTP。對於SCTP流量，建議僅使用靜態網路對象NAT。

問：如果某個節點的塊埠用盡，它是否會丟棄資料包，而不使用下一個可用的IP塊？

不，它不會立即掉下來。它使用來自下一個PAT IP的可用埠塊。如果所有PAT IP上的所有埠塊都已用盡，則會丟棄流量。

問：為了避免集群升級視窗中控制節點的過載，是否最好提前手動選擇新的控制（例如，在4單元集群升級的中途），而不是等待控制節點上處理所有連線？

A.控制元件必須最後更新。這是因為，當控制節點運行較新的版本時，除非所有節點都運行較新的版本，否則它不會啟動池分配。此外，在運行升級時，如果某個控制節點運行的是較舊版本，則其版本較新的所有資料節點都會忽略來自該控制節點的池分發消息。

要詳細解釋這一點，請考慮以4個節點A、B、C和D作為控制節點的群集部署。以下是典型的無中斷升級步驟：

1. 將新版本下載到每個節點。
2. 重新載入裝置「D」。所有連線、xlates都將移動到備份節點。

3. 單位「D」出現，並且：

- a. 處理PAT配置
 - b. 將每個PAT IP分成埠塊
 - c. 使所有埠塊處於未分配狀態
 - d. 忽略從控制元件接收的較舊版本的群集PAT消息
 - e. 將所有PAT連線重定向到主連線。
4. 同樣，使用新版本啟動其他節點。
5. 重新載入裝置'A'控制元件。由於沒有備份用於控制，因此所有現有連線都會被丟棄
6. 新控制元件開始以較新的格式分發埠塊
7. 裝置'A'重新連線，能夠接受埠塊分發消息並對其執行操作

片段處理

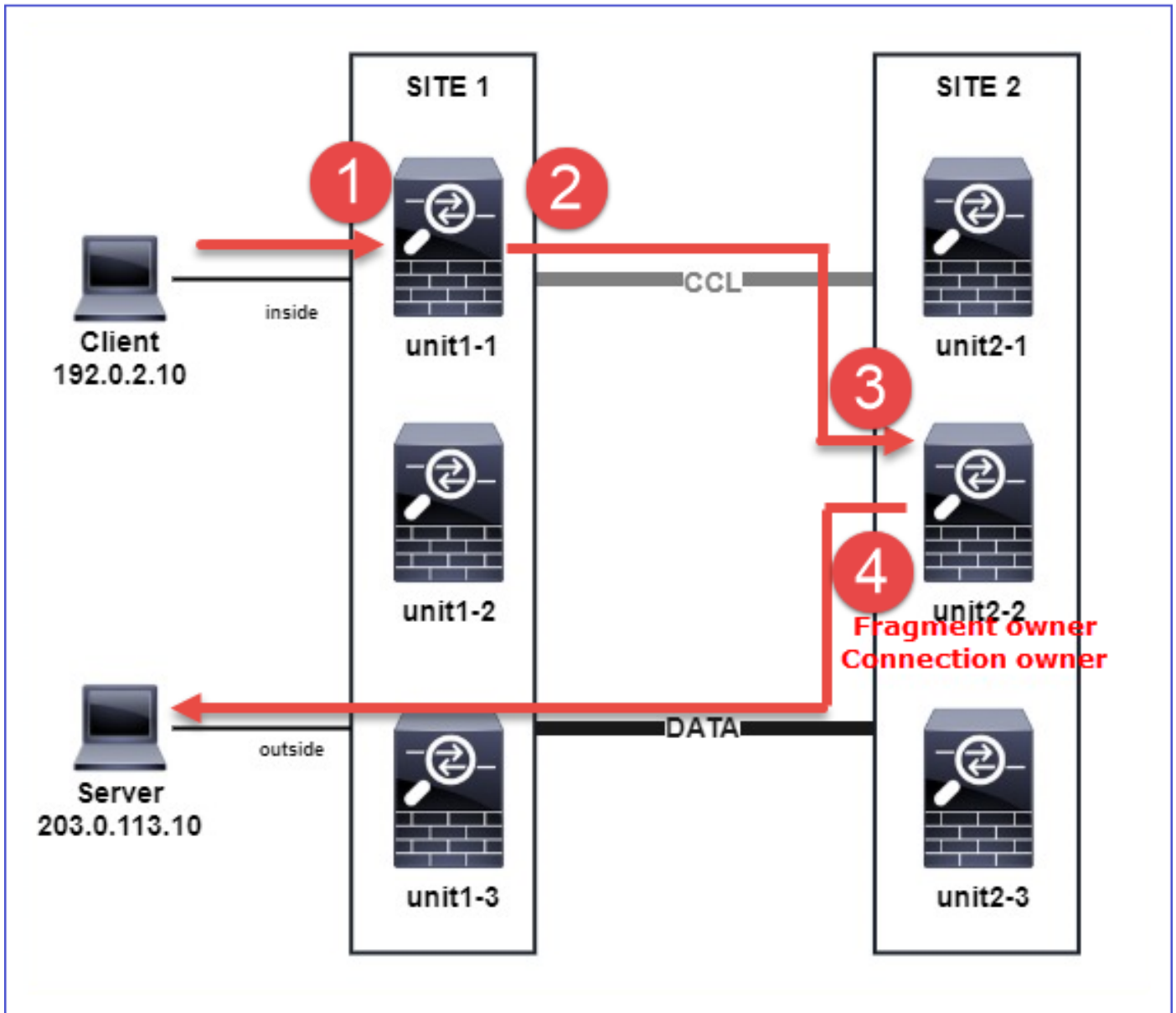
症狀

在站點間群集部署中，必須在一個特定站點（站點本地流量）中處理的分段資料包仍可以傳送到其他站點的裝置，因為這些站點之一可以擁有片段所有者。

在群集邏輯中，為具有分段資料包的連線定義了一個附加角色：片段所有者。

對於分段資料包，接收分段的集群單元根據分段的源IP地址、目標IP地址和資料包ID的雜湊確定分段所有者。然後，所有片段將通過集群控制鏈路轉發給片段所有者。片段可以負載平衡到不同的叢集單元，因為只有第一個片段包含交換器負載平衡雜湊中使用的5元組。其他片段不含來源和目的地連線埠，可以負載平衡到其他叢集裝置。片段所有者臨時重組封包，以便它可以根據來源/目的地IP位址和連線埠的雜湊來判斷導向器。如果是新連線，則片段所有者將成為連線所有者。如果它是現有連線，則片段所有者將通過集群控制鏈路將所有片段轉發給連線所有者。然後，連線所有者重組所有片段。

請考慮使用以下拓撲：從客戶端向伺服器發出分段的ICMP回應請求：



為了瞭解操作的順序，在內部介面、外部介面和集群控制鏈路介面上配置了跟蹤選項，從而捕獲集群範圍的資料包。此外，在內部介面上配置了具有reject-hide選項的資料包捕獲。

```
<#root>
```

```
firepower#
```

```
cluster exec capture capi interface inside trace match icmp any any
```

```
firepower#
```

```
cluster exec capture capir interface inside reinject-hide trace match icmp any any
```

```
firepower#
```

```
cluster exec capture capo interface outside trace match icmp any any
```

```
firepower#
```

```
cluster exec capture capccl interface cluster trace match icmp any any
```

集群內的操作順序：

1. 站點1中的unit-1-1接收分段的ICMP回應請求資料包。

```
<#root>
```

```
firepower#
```

```
cluster exec show cap capir
```

```
unit-1-1(LOCAL)
```

```
:*****
```

```
2 packets captured
```

```
1: 20:13:58.227801 802.1Q vlan#10 P0 192.0.2.10 > 203.0.113.10 icmp: echo request
```

```
2: 20:13:58.227832 802.1Q vlan#10 P0
```

```
2 packets shown
```

2. unit-1-1選擇站點2中的unit-2-2作為片段所有者，並向其傳送分段資料包。

從unit-1-1傳送到unit-2-2的資料包的目標MAC地址是單元-2-2中CCL鏈路的MAC地址。

```
<#root>
```

```
firepower#
```

```
show cap capccl packet-number 1 detail
```

```
7 packets captured
```

```
1: 20:13:58.227817
```

```
0015.c500.018f 0015.c500.029f
```

```
0x0800 Length: 1509
```

```
192.0.2.10 > 203.0.113.10
```

```
icmp: echo request (wrong icmp csum) (frag 46772:1475@0+) (ttl 3)
```

```
1 packet shown
```

```
firepower#
```

```
show cap capccl packet-number 2 detail
```


7 packets captured

2: 20:13:58.227832

0015.c500.018f 0015.c500.029f

0x0800 Length: 637

192.0.2.10 > 203.0.113.10

(

frag 46772

:603@1480) (ttl 3)

1 packet shown

firepower#

cluster exec show interface po48 | i MAC

unit-1-1(LOCAL):*****

MAC address 0015.c500.018f, MTU 1500

unit-1-2:*****

MAC address 0015.c500.019f, MTU 1500

unit-2-2

:*****

MAC address 0015.c500.029f, MTU 1500

unit-1-3:*****

MAC address 0015.c500.016f, MTU 1500

unit-2-1:*****

MAC address 0015.c500.028f, MTU 1500

unit-2-3:*****

MAC address 0015.c500.026f, MTU 1500

3.unit-2-2接收、重組分段的資料包，並成為流的所有者。

<#root>

firepower#

cluster exec unit unit-2-2 show capture capcc1 packet-number 1 trace

11 packets captured

1: 20:13:58.231845 192.0.2.10 > 203.0.113.10 icmp: echo request

Phase: 1

Type: CLUSTER-EVENT

Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'inside'
Flow type: NO FLOW

I (2) received a FWD_FRAG_TO_FRAG_OWNER from (0).

Phase: 2
Type: CLUSTER-EVENT
Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'inside'

Flow type: NO FLOW

I (2) have reassembled a packet and am processing it.

Phase: 3
Type: CAPTURE
Subtype:
Result: ALLOW
Config:
Additional Information:
MAC Access list

Phase: 4
Type: ACCESS-LIST
Subtype:
Result: ALLOW
Config:
Implicit Rule
Additional Information:
MAC Access list

Phase: 5
Type: ROUTE-LOOKUP
Subtype: No ECMP load balancing
Result: ALLOW
Config:
Additional Information:
Destination is locally connected. No ECMP load balancing.
Found next-hop 203.0.113.10 using egress ifc outside(vrfid:0)

Phase: 6
Type: CLUSTER-EVENT
Subtype:
Result: ALLOW
Config:
Additional Information:
Input interface: 'inside'

Flow type: NO FLOW

I (2) am becoming owner

Phase: 7

Type: ACCESS-LIST

Subtype: log

Result: ALLOW

Config:

access-group CSM_FW_ACL_ global

access-list CSM_FW_ACL_ advanced trust ip any any rule-id 268435460 event-log flow-end

access-list CSM_FW_ACL_ remark rule-id 268435460: PREFILTER POLICY: igasimov_prefilter1

access-list CSM_FW_ACL_ remark rule-id 268435460: RULE: r1

Additional Information:

...

Phase: 19

Type: FLOW-CREATION

Subtype:

Result: ALLOW

Config:

Additional Information:

New flow created with id 1719, packet dispatched to next module

...

Result:

input-interface: cluster(vrfid:0)

input-status: up

input-line-status: up

output-interface: outside(vrfid:0)

output-status: up

output-line-status: up

Action: allow

1 packet shown

firepower#

cluster exec unit unit-2-2 show capture capccl packet-number 2 trace

11 packets captured

2: 20:13:58.231875

Phase: 1

Type: CLUSTER-EVENT

Subtype:

Result: ALLOW

Config:

Additional Information:

Input interface: 'inside'

Flow type: NO FLOW

I (2) received a FWD_FRAG_TO_FRAG_OWNER from (0).

Result:

```
input-interface: cluster(vrfid:0)
input-status: up
input-line-status: up
Action: allow
```

1 packet shown

4.unit-2-2根據安全策略允許資料包，並通過外部介面將資料包從站點2傳送到站點1。

<#root>

firepower#

```
cluster exec unit unit-2-2 show cap capo
```

2 packets captured

```
1: 20:13:58.232058 802.1Q vlan#20 P0 192.0.2.10 > 203.0.113.10 icmp: echo request
```

```
2: 20:13:58.232058 802.1Q vlan#20 P0
```

意見/警告

- 與控制器角色不同，片段所有者無法本地化到特定站點中。片段所有者由最初接收新連線的碎片資料包的裝置確定，可以位於任何站點。
- 由於片段所有者也可以成為連線所有者，因此為了將封包轉送到目的地主機，它必須能夠解析輸出介面，並找到目的地主機或下一躍點的IP和MAC位址。這假設下一跳也必須能到達目的地主機。
- 要重組分段的資料包，ASA/FTD會維護每個命名介面的IP分段重組模組。要顯示IP分段重組模組的運算元據，請使用show fragment命令：

<#root>

```
Interface: inside
Configuration:
```

```
size: 200
```

```
, Chain: 24, Timeout: 5, Reassembly: virtual
Run-time stats: Queue: 0, Full assembly: 0
```

```
Drops: Size overflow: 0, Timeout: 0,  
Chain overflow: 0, Fragment queue threshold exceeded: 0,  
Small fragments: 0, Invalid IP len: 0,  
Reassembly overlap: 0, Fraghead alloc failed: 0,  
SGT mismatch: 0, Block alloc failed: 0,  
Invalid IPV6 header: 0, Passenger flow assembly failed: 0
```

在群集部署中，片段所有者或連線所有者將分段的資料包放入片段隊列。片段佇列大小受使用 `fragment size <size> <nameif>` 命令設定的 Size 計數器的值（預設為 200）的限制。當片段佇列大小達到大小的 2/3 時，會視為超出片段佇列閾值，且會捨棄不屬於目前片段鏈一部分的任何新片段。在這種情況下，超出分段隊列閾值將遞增，並生成系統日誌消息 FTD-3-209006。

```
<#root>
```

```
firepower#
```

```
show fragment inside
```

```
Interface: inside
```

```
Configuration:
```

```
Size: 200
```

```
, Chain: 24, Timeout: 5, Reassembly: virtual
```

```
Run-time stats:
```

```
Queue: 133
```

```
, Full assembly: 0
```

```
Drops: Size overflow: 0, Timeout: 8178,
```

```
Chain overflow: 0,
```

```
Fragment queue threshold exceeded: 40802
```

```
,  
Small fragments: 0, Invalid IP len: 0,  
Reassembly overlap: 9673, Fraghead alloc failed: 0,  
SGT mismatch: 0, Block alloc failed: 0,  
Invalid IPV6 header: 0, Passenger flow assembly failed: 0
```

```
%FTD-3-209006: Fragment queue threshold exceeded, dropped TCP fragment from 192.0.2.10/21456 to 203.0.113.1
```

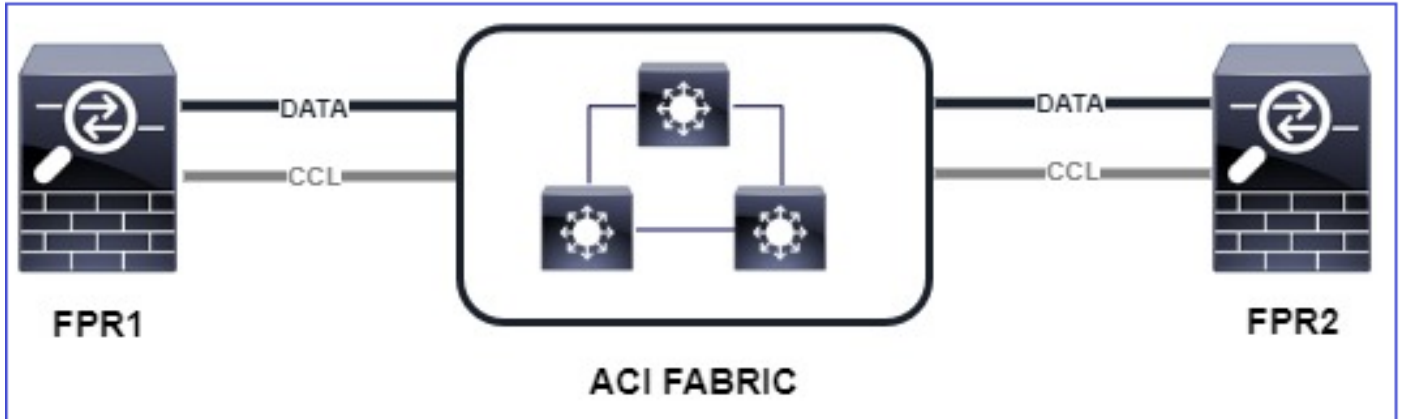
作為解決方法，請在 Firepower Management Center > Devices > Device Management > [Edit Device] > Interfaces > [Interface] > Advanced > Security Configuration > Override Default Fragment Setting 中增加大小，儲存配置和部署策略。然後監控 `show fragment` 命令輸出中的 Queue 計數器以及系統日誌消息 FTD-3-209006 的發生情況。

ACI 問題

由於 ACI Pod 中的活動 L4 校驗和驗證，群集出現間歇性連線問題

症狀

- 通過ACI Pod中部署的ASA/FTD集群出現間歇性連線問題。
- 如果群集中只有1台裝置，則不會出現連線問題。
- 從一個集群單元傳送到集群中一個或多個其他單元的資料包在目標單元的FXOS和資料平面捕獲中不可見。



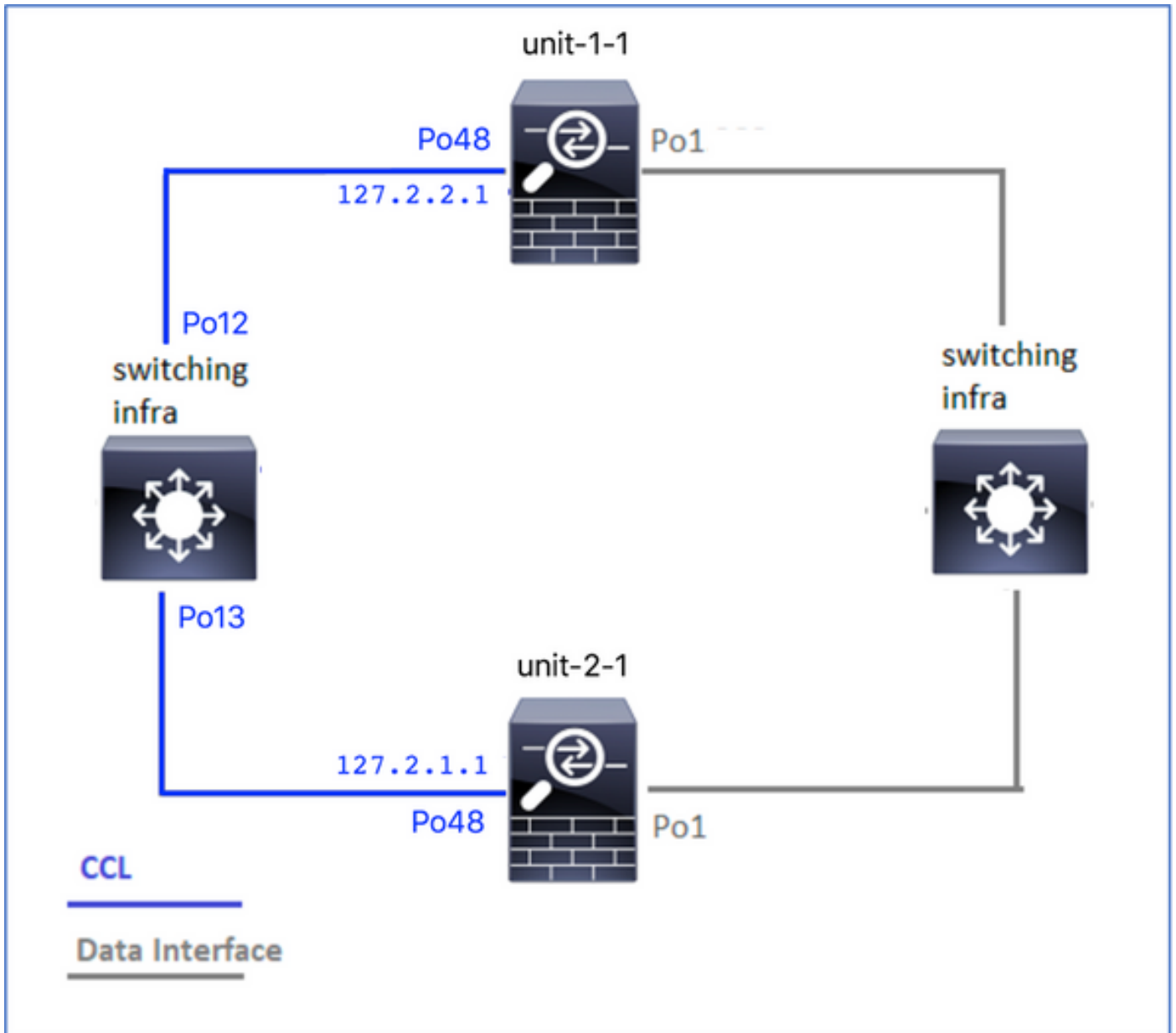
緩解

- 群集控制鏈路上的重定向流量沒有正確的L4校驗和，這是預期行為。群集控制鏈路路徑上的交換機不能驗證L4校驗和。驗證L4校驗和的交換器可能會導致流量遭捨棄。檢查ACI交換矩陣交換機配置，確保未對通過集群控制鏈路收到或傳送的資料包執行L4校驗和。

群集控制平面問題

裝置無法加入群集

CCL上的MTU大小



症狀

裝置無法加入集群，並顯示以下消息：

```
The SECONDARY has left the cluster because application configuration sync is timed out on this unit. Di
Cluster disable is performing cleanup..done.
Unit unit-2-1 is quitting due to system failure for 1 time(s) (last failure is SECONDARY application co
All data interfaces have been shutdown due to clustering being disabled. To recover either enable clust
```

驗證/緩解

- 在FTD上使用show interface指令，確認叢集控制連結介面上的MTU至少比資料介面MTU高100位元組：

<#root>

```
firepower#  
show interface  
  
Interface  
Port-channel1  
"  
Inside  
", is up, line protocol is up  
Hardware is EtherSVI, BW 40000 Mbps, DLY 10 usec  
MAC address 3890.a5f1.aa5e,  
MTU 9084  
  
Interface  
Port-channel48  
"  
cluster  
", is up, line protocol is up  
Hardware is EtherSVI, BW 40000 Mbps, DLY 10 usec  
Description: Clustering Interface  
MAC address 0015.c500.028f,  
MTU 9184  
  
IP address 127.2.2.1, subnet mask 255.255.0.
```

- 使用size選項通過CCL執行ping，以驗證路徑中的所有裝置上CCL MTU上的配置是否正確。

```
<#root>  
firepower#  
ping 127.2.1.1 size 9184
```

- 在交換機上使用show interface命令檢驗MTU配置

```
<#root>  
Switch#  
show interface  
  
port-channel12
```



```
is up
admin state is up,
  Hardware: Port-Channel, address: 7069.5a3a.7976 (bia 7069.5a3a.7976)
```

MTU 9084

bytes, BW 40000000 Kbit , DLY 10 usec

port-channel13

```
is up
admin state is up,
  Hardware: Port-Channel, address: 7069.5a3a.7967 (bia 7069.5a3a.7967)
```

MTU 9084

bytes, BW 40000000 Kbit , DLY 10 use

集群裝置之間的介面不匹配

症狀

裝置無法加入集群，並顯示以下消息：

```
Interface mismatch between cluster primary and joining unit unit-2-1. unit-2-1 aborting cluster join.
Cluster disable is performing cleanup..done.
Unit unit-2-1 is quitting due to system failure for 1 time(s) (last failure is Internal clustering error)
All data interfaces have been shutdown due to clustering being disabled. To recover either enable clustering
```

驗證/緩解

登入到每個機箱上的FCM GUI，導航到Interfaces頁籤，並驗證所有集群成員是否具有相同的介面配置：

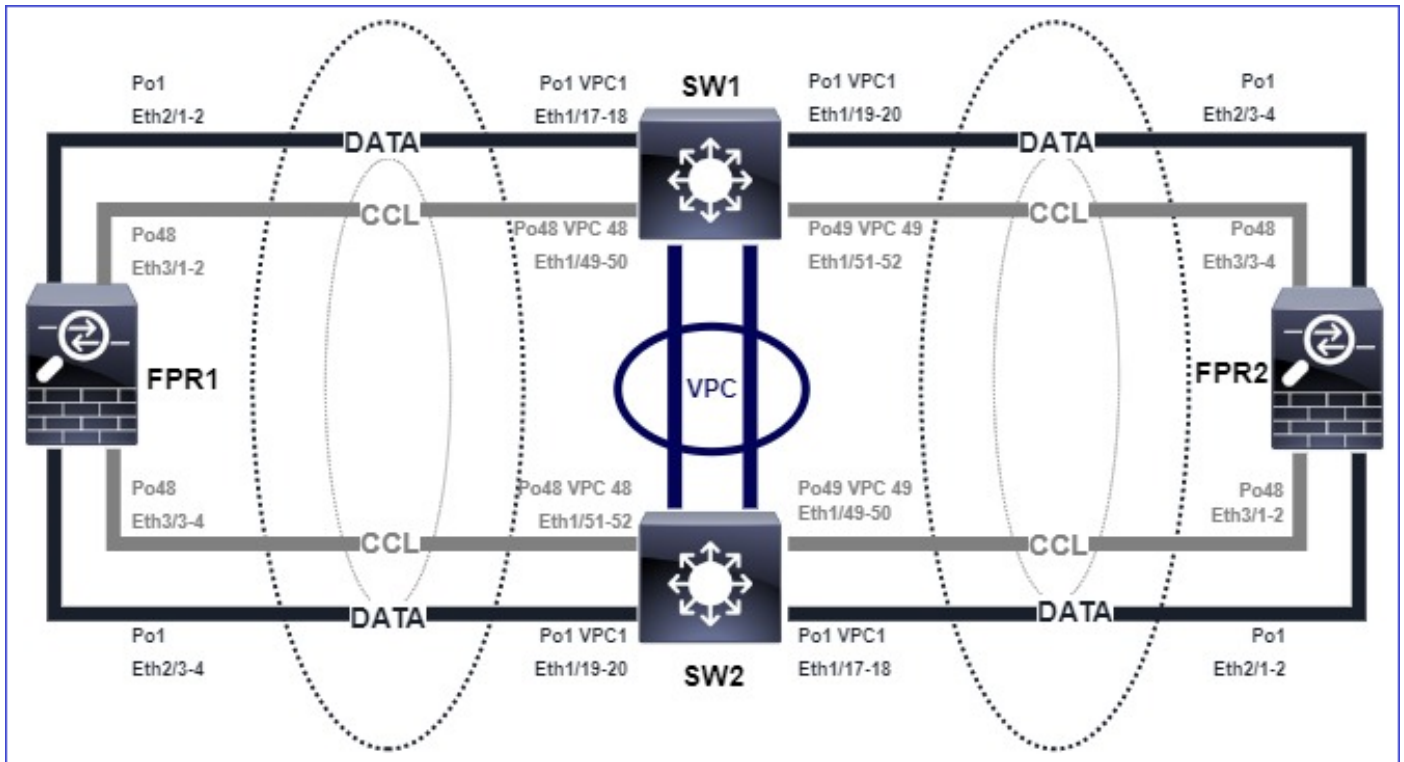
- 分配給邏輯裝置的介面
- 介面的管理速度
- 介面的管理雙工
- 介面狀態

資料/埠通道介面問題

由於在CCL上的可達性問題導致大腦分裂

症狀

群集中有多個控制單元。請考慮使用此拓樸：



機箱1:

<#root>

```
firepower# show cluster info
```

```
Cluster ftd_cluster1: On
Interface mode: spanned
```

```
This is "unit-1-1" in state PRIMARY
```

```
ID : 0
Site ID : 1
Version : 9.15(1)
Serial No.: FLM2103TU5H
CCL IP : 127.2.1.1
CCL MAC : 0015.c500.018f
Last join : 07:30:25 UTC Dec 14 2020
Last leave: N/A
Other members in the cluster:
Unit "unit-1-2" in state SECONDARY
ID : 1
Site ID : 1
Version : 9.15(1)
Serial No.: FLM2103TU4D
CCL IP : 127.2.1.2
CCL MAC : 0015.c500.019f
Last join : 07:30:26 UTC Dec 14 2020
Last leave: N/A
Unit "unit-1-3" in state SECONDARY
ID : 3
Site ID : 1
Version : 9.15(1)
```

Serial No.: FLM2102THJT
CCL IP : 127.2.1.3
CCL MAC : 0015.c500.016f
Last join : 07:31:49 UTC Dec 14 2020
Last leave: N/A

機箱2:

<#root>

```
firepower# show cluster info
```

Cluster ftd_cluster1: On
Interface mode: spanned

This is "unit-2-1" in state PRIMARY

ID : 4
Site ID : 1
Version : 9.15(1)
Serial No.: FLM2103TUN1
CCL IP : 127.2.2.1
CCL MAC : 0015.c500.028f
Last join : 11:21:56 UTC Dec 23 2020
Last leave: 11:18:51 UTC Dec 23 2020
Other members in the cluster:
Unit "unit-2-2" in state SECONDARY
ID : 2
Site ID : 1
Version : 9.15(1)
Serial No.: FLM2102THR9
CCL IP : 127.2.2.2
CCL MAC : 0015.c500.029f
Last join : 11:18:58 UTC Dec 23 2020
Last leave: 22:28:01 UTC Dec 22 2020
Unit "unit-2-3" in state SECONDARY
ID : 5
Site ID : 1
Version : 9.15(1)
Serial No.: FLM2103TUM1
CCL IP : 127.2.2.3
CCL MAC : 0015.c500.026f
Last join : 11:20:26 UTC Dec 23 2020
Last leave: 22:28:00 UTC Dec 22 2020

驗證

- 使用ping命令驗證控制單元的集群控制鏈路(CCL)IP地址之間的連通性：

<#root>

```
firepower# ping 127.2.1.1
```

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 127.2.1.1, timeout is 2 seconds:

?????

Success rate is 0 percent (0/5)

- 檢查ARP表：

```
<#root>
```

```
firepower# show arp
```

```
cluster 127.2.2.3 0015.c500.026f 1
```

```
cluster 127.2.2.2 0015.c500.029f 1
```

- 在控制單元中，配置並檢查CCL介面上的捕獲：

```
<#root>
```

```
firepower# capture capccl interface cluster
```

```
firepower# show capture capccl | i 127.2.1.1
```

```
2: 12:10:57.652310 arp who-has 127.2.1.1 tell 127.2.2.1
41: 12:11:02.652859 arp who-has 127.2.1.1 tell 127.2.2.1
74: 12:11:07.653439 arp who-has 127.2.1.1 tell 127.2.2.1
97: 12:11:12.654018 arp who-has 127.2.1.1 tell 127.2.2.1
126: 12:11:17.654568 arp who-has 127.2.1.1 tell 127.2.2.1
151: 12:11:22.655148 arp who-has 127.2.1.1 tell 127.2.2.1
174: 12:11:27.655697 arp who-has 127.2.1.1 tell 127.2.2.1
```

緩解

- 確保CCL埠通道介面已連線到交換機上的獨立埠通道介面。
- 在Nexus交換機上使用虛擬埠通道(vPC)時，請確保CCL埠通道介面連線到不同的vPC，並且vPC配置沒有失敗的一致性狀態。
- 確保CCL埠通道介面位於同一個廣播域中，並且已在介面上建立並允許CCL VLAN。

以下是交換器組態範例：

```
<#root>
```

```
Nexus#
```

```
show run int po48-49
```

```
interface port-channel48
description FPR1
```

```
switchport access vlan 48
```

```
vpc 48
```

```
interface port-channel49
description FPR2
```

```
switchport access vlan 48
```

```
vpc 49
```

```
Nexus#
```

```
show vlan id 48
```

```
VLAN Name Status Ports
```

```
-----
```

```
48 CCL active Po48, Po49, Po100, Eth1/53, Eth1/54
```

```
VLAN Type Vlan-mode
```

```
-----
```

```
48 enet CE
```

```
1 Po1 up success success 10,20
```

```
48 Po48 up success success 48
```

```
49 Po49 up success success 48
```

```
<#root>
```

```
Nexus1#
```

```
show vpc brief
```

```
Legend:
```

(*) - local vPC is down, forwarding via vPC peer-link

vPC domain id : 1
Peer status : peer adjacency formed ok
vPC keep-alive status : peer is alive
Configuration consistency status : success

Per-vlan consistency status : success

Type-2 consistency status : success

vPC role : primary
Number of vPCs configured : 3
Peer Gateway : Disabled
Dual-active excluded VLANs : -
Graceful Consistency Check : Enabled
Auto-recovery status : Disabled
Delay-restore status : Timer is off.(timeout = 30s)
Delay-restore SVI status : Timer is off.(timeout = 10s)

vPC Peer-link status

id Port Status Active vlans

1 Po100 up 1,10,20,48-49,148

vPC status

id Port Status Consistency Reason Active vlans

1 Po1 up success success 10,20

48 Po48 up success success 48

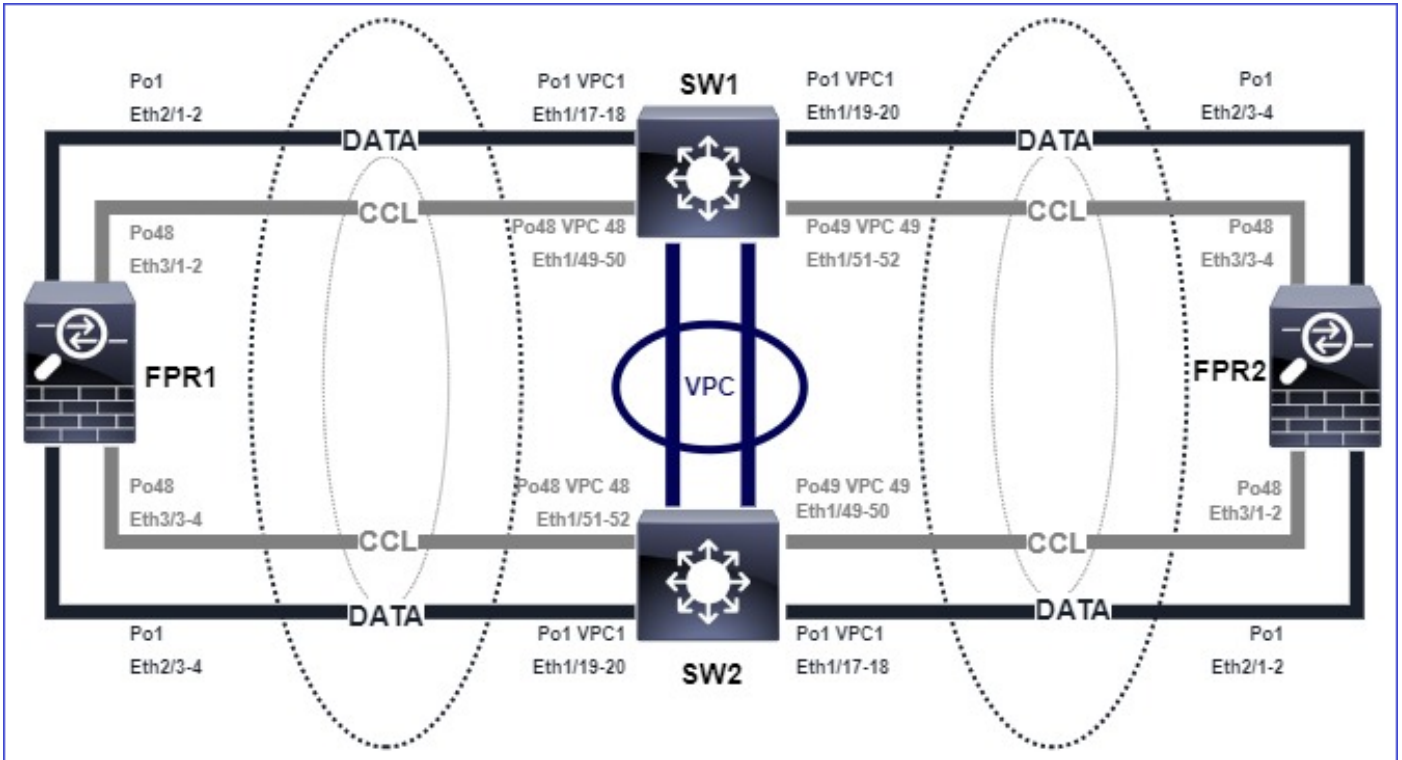
49 Po49 up success success 48

由於暫停的資料埠通道介面而禁用群集

症狀

一個或多個資料埠通道介面被掛起。當管理性啟用的資料介面掛起時，由於介面運行狀況檢查失敗，同一機箱中的所有群集單元都將從群集中彈出。

請考慮使用此拓樸：



驗證

- 檢查控制單元控制檯：

```
<#root>
```

```
firepower#
```

```
Beginning configuration replication to
```

```
SECONDARY unit-2-2
```

```
End Configuration Replication to SECONDARY.
```

```
Asking SECONDARY unit
```

```
unit-2-2
```

```
to quit because it
```

```
failed interface health
```

```
check 4 times (last failure on
```

```
Port-channel1
```

```
). Clustering must be manually enabled on the unit to rejoin.
```

- 檢查受影響裝置中show cluster history和show cluster info trace module hc命令的輸出：

```
<#root>
```

```
firepower# Unit is kicked out from cluster because of interface health check failure.
```

```
Cluster disable is performing cleanup..done.
```

All data interfaces have been shutdown due to clustering being disabled. To recover either enable clust

Cluster unit unit-2-1 transitioned from SECONDARY to DISABLED

firepower#

```
show cluster history
```

```
=====
From State To State Reason
=====
```

12:59:37 UTC Dec 23 2020

ONCALL SECONDARY_COLD Received cluster control message

12:59:37 UTC Dec 23 2020

SECONDARY_COLD SECONDARY_APP_SYNC Client progression done

13:00:23 UTC Dec 23 2020

SECONDARY_APP_SYNC SECONDARY_CONFIG SECONDARY application configuration sync done

13:00:35 UTC Dec 23 2020

SECONDARY_CONFIG SECONDARY_FILESYS Configuration replication finished

13:00:36 UTC Dec 23 2020

SECONDARY_FILESYS SECONDARY_BULK_SYNC Client progression done

13:01:35 UTC Dec 23 2020

SECONDARY_BULK_SYNC DISABLED Received control message DISABLE (interface health check failure)

<#root>

firepower#

```
show cluster info trace module hc
```

Dec 23 13:01:36.636 [INFO]cluster_fsm_clear_np_flows: The clustering re-enable timer is started to expi

Dec 23 13:01:32.115 [INFO]cluster_fsm_disable: The clustering re-enable timer is stopped.

Dec 23 13:01:32.115 [INFO]Interface Port-channel1 is down

- 在fxos命令外殼中檢查show port-channel summary命令的輸出：

<#root>

FPR2(fxos)#

```
show port-channel summary
```


Flags: D - Down P - Up in port-channel (members)
I - Individual H - Hot-standby (LACP only)
s - Suspended r - Module-removed
S - Switched R - Routed
U - Up (port-channel)
M - Not in use. Min-links not met

Group Port-Channel Type Protocol Member Ports

1 Po1(SD) Eth LACP Eth2/1(s) Eth2/2(s) Eth2/3(s) Eth2/4(s)

48 Po48(SU) Eth LACP Eth3/1(P) Eth3/2(P) Eth3/3(P) Eth3/4(P)

緩解

- 確保所有機箱具有相同的群集組名稱和密碼。
- 確保所有機箱和交換機中的埠通道介面具有相同的雙工/速度配置，並已管理性啟用物理成員介面。
- 在站點內集群中，確保所有機箱中的相同資料埠通道介面連線到交換機上的相同埠通道介面。
- 在Nexus交換機中使用虛擬埠通道(vPC)時，請確保vPC配置沒有失敗的一致性狀態。
- 在站點內集群中，確保所有機箱中的相同資料埠通道介面連線到同一個vPC。

群集穩定性問題

FXOS回溯

症狀

裝置離開集群。

驗證/緩解

- 使用show cluster history命令檢視設備何時離開集群

```
<#root>
```

```
firepower#
```

```
show cluster history
```

- 使用這些命令檢查FXOS是否有回溯

```
<#root>
```

```
FPR4150#
```

```
connect local-mgmt
```

```
FPR4150 (local-mgmt)#
```

```
dir cores
```

- 收集裝置離開集群時生成的核心檔案，並將其提供給TAC。

磁碟已滿

如果群集單元的/ngfw分割槽中的磁碟利用率達到94%，則該單元會退出群集。每3秒進行一次磁碟利用率檢查：

```
<#root>
```

```
> show disk
```

```
Filesystem Size Used Avail Use% Mounted on
rootfs 81G 421M 80G 1% /
devtmpfs 81G 1.9G 79G 3% /dev
tmpfs 94G 1.8M 94G 1% /run
tmpfs 94G 2.2M 94G 1% /var/volatile
/dev/sda1 1.5G 156M 1.4G 11% /mnt/boot
/dev/sda2 978M 28M 900M 3% /opt/cisco/config
/dev/sda3 4.6G 88M 4.2G 3% /opt/cisco/platform/logs
/dev/sda5 50G 52M 47G 1% /var/data/cores
/dev/sda6 191G 191G 13M
```

```
100% /ngfw
```

```
cgroup_root 94G 0 94G 0% /dev/cgroups
```

在這種情況下，show cluster history輸出顯示：

```
<#root>
```

```
15:36:10 UTC May 19 2021
```

```
PRIMARY Event: Primary unit unit-1-1 is quitting
                due to
```

```
diskstatus
```

```
Application health check failure, and
                primary's application state is down
```

或

14:07:26 CEST May 18 2021

SECONDARY DISABLED Received control message DISABLE (application health check failure)

驗證故障的另一種方法是：

```
<#root>
```

```
firepower#
```

```
show cluster info health
```

```
Member ID to name mapping:
```

```
0 - unit-1-1(myself) 1 - unit-2-1
```

```
          0  1
Port-channel48 up up
Ethernet1/1 up up
Port-channel12 up up
Port-channel13 up up
```

```
Unit overall          healthy healthy
```

```
Service health status:
```

```
          0      1
```

```
diskstatus (monitor on) down down
```

```
snort (monitor on)      up      up
```

```
Cluster overall        healthy
```

此外，如果磁碟大約是100%，則裝置在釋放一些磁碟空間之前可能難以重新加入群集。

溢位保護

每隔5分鐘，每個集群單元檢查本地和對等單元的CPU和記憶體利用率。如果利用率高於系統閾值（LINA CPU 50%或LINA記憶體59%），資訊性消息將顯示在：

- 系統日誌(FTD-6-748008)
- 檔案log/cluster_trace.log，例如：

```
<#root>
```

```
firepower#
```

```
more log/cluster_trace.log | i CPU
```

```
May 20 16:18:06.614 [INFO][
```

```
CPU load 87%
```

```
| memory load 37%] of module 1 in chassis 1 (unit-1-1) exceeds overflow protection threshold [
```

] . System may be oversubscribed on member failure.

May 20 16:18:06.614 [INFO][CPU load 87% | memory load 37%] of chassis 1 exceeds overflow protection thr

May 20 16:23:06.644 [INFO][CPU load 84% | memory load 35%] of module 1 in chassis 1 (unit-1-1) exceeds

該消息表示在單元出現故障時，其他單元資源可以超額訂閱。

簡化模式


6.3之前FMC版本的行為

- 您可以在FMC上單獨註冊每個群集節點。
- 然後，在FMC中形成邏輯群集。
- 對於每次新增新的群集節點，您必須手動註冊該節點。

6.3後FMC

- 簡化模式功能允許您一步在FMC上註冊整個集群（只需註冊集群的任何一個節點）。

最低支援的管理器	受管裝置	需要支援的最低受管裝置版本	備註
FMC 6.3	僅限FP9300和FP4100上的FTD叢集	6.2.0	這僅是FMC功能

 **警告：** 在FTD上建立叢集後，需要等待自動註冊啟動。您不能嘗試手動註冊群集節點（新增裝置），但使用Reconcile選項。

症狀

節點註冊失敗

- 如果控制節點註冊由於任何原因失敗，則從FMC中刪除該集群。

緩解

如果資料節點註冊因任何原因而失敗，則有2個選項：

1. 在群集的每個部署中，FMC檢查是否有需要註冊的群集節點，然後啟動這些節點的自動註冊。
2. 群集摘要頁籤下有一個Reconcile選項(Devices > Device Management > Cluster tab > View Cluster Status連結)。觸發協調操作後，FMC開始自動註冊需要註冊的節點。

相關資訊

- [用於Firepower威脅防禦的群集](#)
- [適用於Firepower 4100/9300機箱的ASA群集](#)
- [關於Firepower 4100/9300機箱上的群集](#)
- [Firepower NGFW群集深入探討 — BRKSEC-3032](#)
- [分析 Firepower 防火牆擷取，以有效針對網路問題進行疑難排解](#)

關於此翻譯

思科已使用電腦和人工技術翻譯本文件，讓全世界的使用者能夠以自己的語言理解支援內容。請注意，即使是最佳機器翻譯，也不如專業譯者翻譯的內容準確。Cisco Systems, Inc. 對這些翻譯的準確度概不負責，並建議一律查看原始英文文件（提供連結）。