ACI安全策略故障排除 — 合約

目錄

簡介

本文檔介紹瞭解並排除ACI安全策略（稱為合約）故障的步驟。

背景資訊

本文檔中的資料摘自故障排除思科以應用為中心的基礎設施，第二版書，特別是Security Policies - Overview、Security Policies - Tools、**Security Policies - EPG到EPG**、**Security Policies - Preferred group和**Security Policies - vzAny to EPG章。

概觀

ACI解決方案的基本安全架構遵循許可清單模式。除非在**unenforced**模式下配置VRF，否則所有EPG到EPG流量都會隱式丟棄。正如預置許可清單模型所暗示的那樣，預設VRF設定處於**強制**模式。通過在交換機節點上實施分割槽規則，可以允許或明確拒絕流量。根據終端組(EPG)之間的期望通訊流和用於定義它們的方法，這些分割槽規則可以程式設計為各種不同的配置。請注意，分割槽規則項不具有狀態性，通常會在規則程式設計後基於埠/套接字給定兩個EPG允許/拒絕。

分割槽規則的程式設計方法

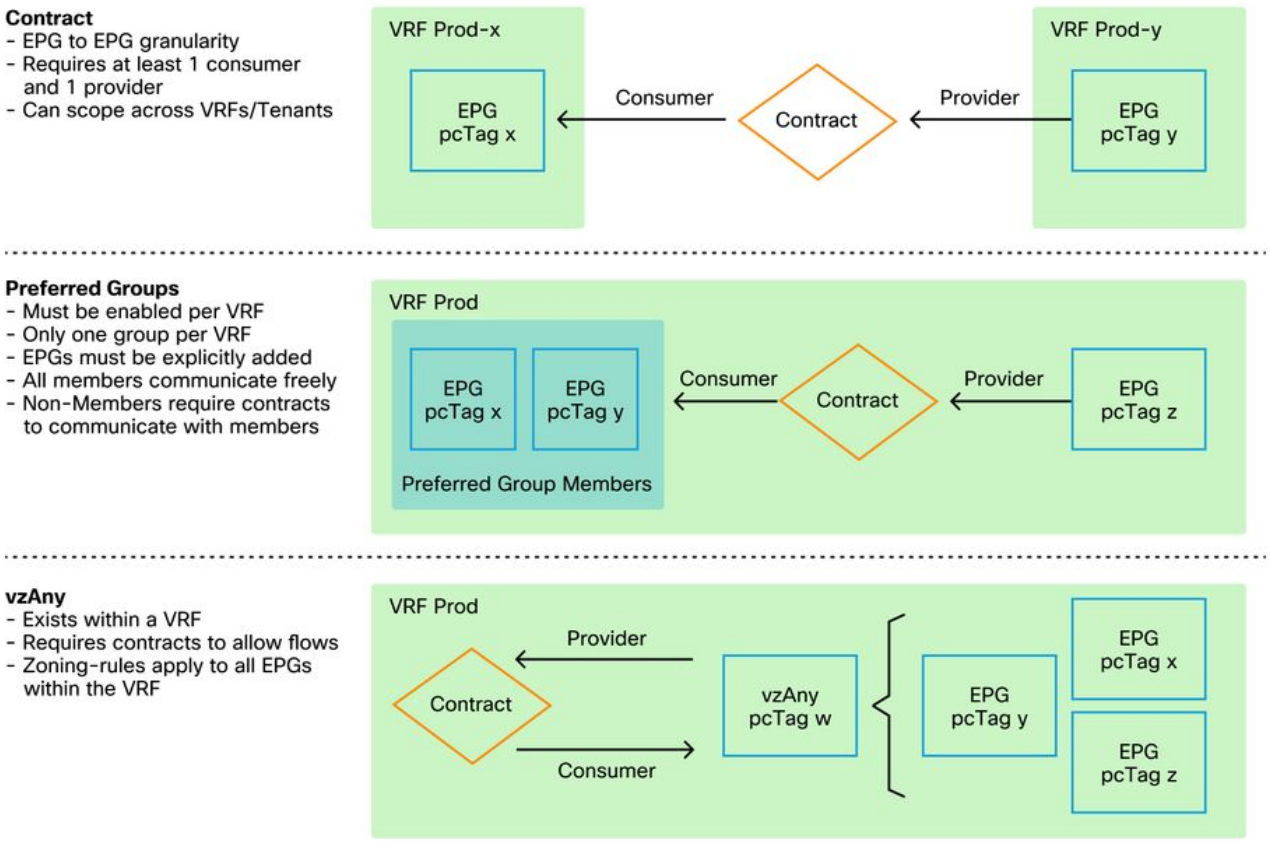
對ACI中的分割槽規則進行程式設計的主要方法如下：

- **EPG到EPG合約：通常**至少需要一個消費者和一個提供商來跨兩個或多個不同的終端組規劃分割槽規則。

- **首選組：需**要在VRF級別啟用分組；每個VRF只能存在一個組。小組所有成員均可自由交流。非成員需要合約以允許流向首選組。
- **vzAny:**在給定VRF下定義的「EPG集合」。vzAny表示VRF中的所有EPG。使用vzAny允許一個EPG和VRF內的所有EPG之間通過一個合約連線進行流動。

以下圖表可用於引用上述每種方法允許控制的分割槽規則的粒度：

分割槽規則方法的比較



在利用規劃分割槽規則的合約方法時，有一個定義合約範圍的選項。如果需要任何路由洩漏/共用服務設計，必須仔細考慮此選項。如果希望在ACI交換矩陣內從一個VRF連線到另一個VRF，則使用合約進行此操作。

範圍值可以是以下值：

- **應用產品**：合約使用者/提供者關係將僅對同一應用配置檔案內定義的EPG之間的規則進行程式設計。在其它應用配置檔案EPG之間重複使用相同的合約將不允許它們之間的串擾。
- **VRF（預設）：合**同使用者/提供商關係將程式設計在同一VRF中定義的EPG之間的規則。在其他應用配置檔案EPG上重複使用相同的合約會允許它們之間的串擾。注意確保只允許所需的流，否則應定義新的合約以防止無意串擾。
- **租戶：合**同使用者/提供商關係將在同一租戶內定義的EPG之間規劃規則。如果在單個租戶內存在與多個VRF繫結的EPG，並且它們使用/提供相同的合約，則此範圍可用於誘導路由洩漏，以允許VRF間通訊。
- **全域性：合**同消費者/提供商關係將對ACI交換矩陣內任何租戶之間的EPG之間的規則進行程式設計。這是該定義的最高範圍，在先前定義的合約上啟用該定義時應格外小心，以防止無意的流量洩漏。

讀取分割槽規則條目

一旦對分割槽規則進行了程式設計，它將在枝葉上顯示如下：

```
+---------+--------+--------+----------+---------------+---------+---------+----------------+-
---------+---------------------+
| Rule ID | SrcEPG | DstEPG | FilterID |      Dir      |  operSt |  Scope  |       Name     |
Action  |       Priority      |
+---------+--------+--------+----------+---------------+---------+---------+----------------+-
---------+---------------------+
```

- **規則ID**：規則條目的ID。除了充當唯一識別符號之外，沒有真正意義。
- **源EPG:**源終端組的每個VRF(pcTag)的唯一ID。
- **Dst EPG：目**的地終端組的每個VRF(pcTag)的唯一ID。
- **FilterID:**規則嘗試與之匹配的過濾器的ID。過濾器包含規則匹配的協定資訊。
- **Dir:**zoning-rule的方向性。
- **OperSt:**規則的操作狀態。
- **範圍**：規則將匹配的VRF的唯一ID。
- **名稱**：導致該條目被程式設計的合約的名稱。
- **Action:**葉與該條目匹配時將執行的操作。包括：[Drop， Permit， Log， Redirect]。
- **優先順序**：在給定匹配範圍、SrcEPG、DstEPG和篩選器條目的情況下驗證分割槽規則進行操作的順序。

原則內容可定址記憶體(CAM)

隨著每個分割槽規則的程式設計，根據過濾器條目對映的分割槽規則條目的矩陣將開始使用交換機上的**策略CAM**。在設計通過ACI交換矩陣的允許流時，應特別注意重複使用合約，而不是根據最終設計建立新合約。在不瞭解結果的分割槽規則的情況下，隨意地跨多個EPG重複使用同一合約，可能會快速級聯成多個意外允許的流。同時，這些無意的流量將繼續消耗策略CAM。當策略CAM滿時，分割槽規則程式設計將開始失敗，這可能會導致意外和間歇性丟失，具體取決於配置和終端行為。

共用L3Outs的VRF洩漏、全域性pcTags和策略實施方向

這是需要配置合約的共用服務使用案例的特殊標註。共用服務通常意味著ACI交換矩陣內的VRF間流量，該流量依賴於使用「租戶」或「全域性」範圍的合約。要充分理解這一點，首先必須強化以下觀點：分配給EPG的典型pcTag值並非全球唯一。pcTag的範圍被限定在VRF中，並且同一個pcTag可能會在另一個VRF中重複使用。當討論路由洩漏時，開始在ACI交換矩陣上實施要求，包括需要全域性唯一值，包括子網和pcTags。

這一點之所以成為一種特殊考慮，是因為與EPG消費者與提供商相關的方向性方面。在共用服務方案中，通常希望提供商驅動全域性pcTag以獲得交換矩陣唯一值。同時，消費者將保留其VRF範圍的pcTag，使其處於特殊位置，以便現在能夠程式設計和瞭解使用全域性pcTag值來實施策略。

作為參考，pcTag分配範圍如下：

- 系統保留：1-15 .
- 全域性範圍：共用服務16384供商EPG採用16-GBPS。
- 本地作用域：16385-65535 for VRF scoped EPGs。

VRF策略控制實施方向

在每個VRF中，可以定義實施方向設定。

- 實施方向的預設設定為Ingress。
- 實施方向的另一個選項是Egress。

瞭解策略的實施位置取決於幾個不同的變數。

下表有助於瞭解在枝葉級別實施安全策略的位置。

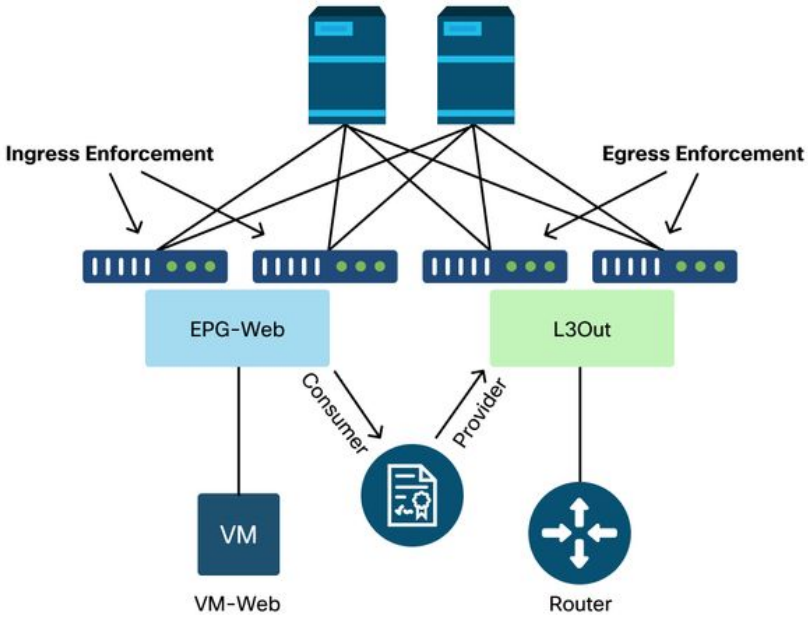
在哪裡執行策略？

案例	VRF實施模式	消費者	提供商	策略實施於
VRF內	輸入/輸出	EPG	EPG	·如果獲知目標端點：入口枝葉* ·如果未獲知目標終結點：出口分葉
	輸入	EPG	L3Out EPG	消費者枝葉（非邊界枝葉）
	輸入	L3Out EPG	EPG	提供程式分葉（非邊界分葉）
	輸出	EPG	L3Out EPG	邊界枝葉 — >非邊界枝葉流量 ·如果獲知目標端點：邊界枝葉
	輸出	L3Out EPG	EPG	·如果未獲知目標終結點：非邊界枝葉 非邊界枝葉 — >邊界枝葉流量 ·邊界枝葉
	輸入/輸出	L3Out EPG	L3Out EPG	入口枝葉*
	輸入/輸出	EPG	EPG	消費者枝葉
	輸入/輸出	EPG	L3Out EPG	消費者枝葉（非邊界枝葉）
VRF間	輸入/輸出	L3Out EPG	EPG	入口枝葉*
	輸入/輸出	L3Out EPG	L3Out EPG	入口枝葉*

*策略實施應用於資料包所命中的第一個枝葉。

下圖顯示合約實施的示例，其中EPG-Web作為消費者，L3Out EPG作為提供商具有VRF內合約。如果VRF設定為入口實施模式，則由EPG-Web所在的枝葉節點實施策略。如果VRF設定為出口實施模式，則如果在邊界枝葉上獲取VM-Web端點，則由L3Out所在的邊界枝葉節點實施策略。

輸入強制和輸出強制執行



工具

有許多工具和命令可用於幫助識別策略**丟棄**。策略丟棄可定義為由於合約配置或缺少合約配置而導致的丟包。

Zoning-rule validation

以下工具和命令可用於明確驗證由於已完成的合約使用者/提供商關係而在枝葉交換機上程式設計的分割槽規則。

'show zoning-rules'

顯示所有分割槽規則的交換機級別命令。

```
leaf# show zoning-rule
+---------+--------+--------+----------+----------------+---------+----------+----------------
+----------+---------------------------+
| Rule ID | SrcEPG | DstEPG | FilterID |      Dir       |  operSt |  Scope   |       Name
|  Action  |          Priority          |
+---------+--------+--------+----------+----------------+---------+----------+----------------
+----------+---------------------------+
|   4156  |   25   | 16410  |   425    | uni-dir-ignore | enabled | 2818048  | external_to_ntp
|  permit  |       fully_qual(7)        |
|   4131  | 16410  |   25   |   424    |     bi-dir     | enabled | 2818048  | external_to_ntp
|  permit  |       fully_qual(7)        |
+---------+--------+--------+----------+----------------+---------+----------+----------------
+----------+---------------------------+
```

'show zoning-filter'

包含分割槽規則所執行的運動/資料流的篩選器。過濾器程式設計可以用此命令驗證。

```
leaf# show zoning-filter
+----------+----------+-------------+-------------+-------------+----------+-------------+------
-------+-------------+-------------+----------+
| FilterId |   Name   |    EtherT   |     Prot    | ApplyToFrag | Stateful |  SFromPort  |
SToPort   |  DFromPort  |   DToPort   |   Prio   |
+----------+----------+-------------+-------------+-------------+----------+-------------+------
-------+-------------+-------------+----------+
| implarp  | implarp  |     arp     | unspecified |      no     |    no    | unspecified |
unspecified | unspecified | unspecified |  dport   |
| implicit | implicit | unspecified | unspecified |      no     |    no    | unspecified |
unspecified | unspecified | unspecified | implicit |
|   425    |  425_0   |      ip     |     tcp     |      no     |    no    |     123     |
123     | unspecified | unspecified |  sport   |
|   424    |  424_0   |      ip     |     tcp     |      no     |    no    | unspecified |
unspecified |     123     |     123     |  dport   |
+----------+----------+-------------+-------------+-------------+----------+-------------+------
-------+-------------+-------------+----------+
```

'顯示系統內部policy-mgr統計資訊'

可以運行此命令來驗證每個分割槽規則的命中數。這對於確定預期規則相對於另一個規則是否被命中很有用，例如可能具有更高優先順序的隱式丟棄規則。

```
leaf# show system internal policy-mgr stats
Requested Rule Statistics
Rule (4131) DN (sys/actrl/scope-2818048/rule-2818048-s-16410-d-25-f-424) Ingress: 0, Egress: 0,
Pkts: 0  RevPkts: 0
Rule (4156) DN (sys/actrl/scope-2818048/rule-2818048-s-25-d-16410-f-425) Ingress: 0, Egress: 0,
Pkts: 0  RevPkts: 0
```

'show logging ip access-list internal packet-log deny'

可以在iBash級別運行的交換機級別命令，該命令報告ACL（合約）相關的丟棄和流量相關資訊，包括：

- VRF
- VLAN-ID
- 源MAC/目標MAC
- 源IP/目標IP
- 來源連線埠/目的地連線埠
- 來源介面

```
leaf# show logging ip access-list internal packet-log deny
[ Tue Oct  1 10:34:37 2019 377572 usecs]: CName: Prod1:VRF1(VXLAN: 2654209), VlanType: Unknown,
Vlan-Id: 0, SMac: 0x000c0c0c0c0c, DMac:0x000c0c0c0c0c, SIP: 192.168.21.11, DIP: 192.168.22.11,
SPort: 0, DPort: 0, Src Intf: Tunnel7, Proto: 1, PktLen: 98
[ Tue Oct  1 10:34:36 2019 377731 usecs]: CName: Prod1:VRF1(VXLAN: 2654209), VlanType: Unknown,
Vlan-Id: 0, SMac: 0x000c0c0c0c0c, DMac:0x000c0c0c0c0c, SIP: 192.168.21.11, DIP: 192.168.22.11,
SPort: 0, DPort: 0, Src Intf: Tunnel7, Proto: 1, PktLen: 98
```

contract_parser

一種裝置上的Python指令碼，它根據ID執行名稱查詢時，生成一個輸出，該輸出將分割槽規則、過濾器和命中統計資訊相關聯。此指令碼非常有用，因為它採用多步驟流程，並將其轉換為單個命令，該命令可以過濾到特定EPG/VRF或其他合約相關值。

```
leaf# contract_parser.py
Key:
[prio:RuleId] [vrf:{str}] action protocol src-epg [src-l4] dst-epg [dst-l4]
[flags][contract:{str}] [hit=count]

[7:4131] [vrf:common:default] permit ip tcp tn-Prod1/ap-Services/epg-NTP(16410) tn-Prod1/l3out-
L3Out1/instP-extEpg(25) eq 123  [contract:uni/tn-Prod1/brc-external_to_ntp] [hit=0]
[7:4156] [vrf:common:default] permit ip tcp tn-Prod1/l3out-L3Out1/instP-extEpg(25) eq 123 tn-
Prod1/ap-Services/epg-NTP(16410)  [contract:uni/tn-Prod1/brc-external_to_ntp] [hit=0]
[12:4169] [vrf:common:default] deny,log any tn-Prod1/l3out-L3Out1/instP-extEpg(25) epg:any
[contract:implicit] [hit=0]
[16:4167] [vrf:common:default] permit any epg:any tn-Prod1/bd-Services(32789)
[contract:implicit] [hit=0]
```

資料包分類驗證

伊拉姆語

用於檢查轉發詳細資訊的ASIC級別報告，在丟包的情況下表明丟棄原因。與此部分相關，原因可能是SECURITY_GROUP_DENY（合約策略刪除）。

分類

APIC上基於Python的實用程式，它可使用ELAM跟蹤端到端資料包流。

ELAM助理應用

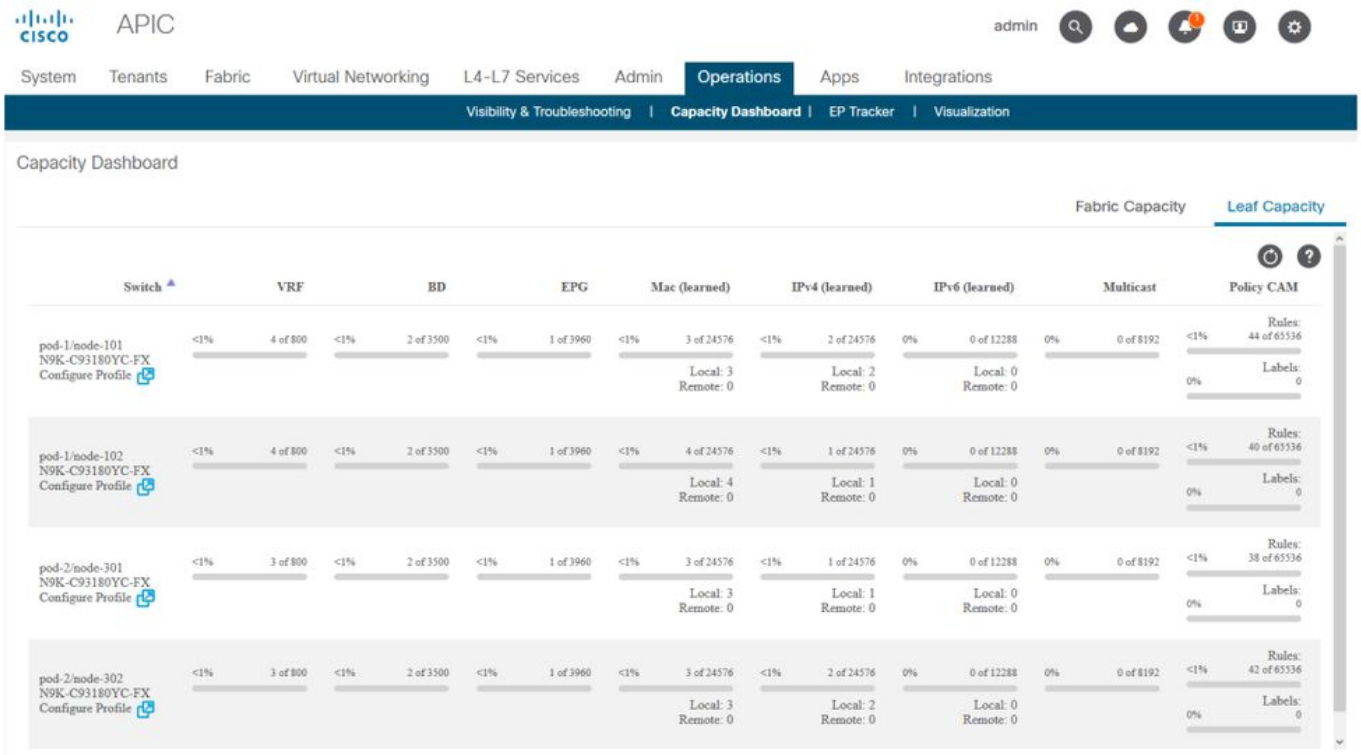
一個APIC應用，它抽象化各種ASIC的複雜性，使轉發決策檢查更加方便和使用者友好。

有關ELAM、Triage和ELAM Assistant工具的其他詳細資訊，請參閱「交換矩陣內轉發」部分

策略CAM使用情況

基於每個枝葉的策略CAM使用量是一個重要的監控引數，以確保交換矩陣處於健康狀態。最快的監控方法是使用GUI中的「Capacity Dashboard」並明確檢查「Policy Cam」列。

容量控制面板的「枝葉容量」檢視



'show platform internal hal health-stats'

此命令對於驗證各種資源限制和使用情況非常有用，包括策略CAM。請注意，此命令只能在vsh_lc中運行，因此如果從iBash運行，請使用「 — c」標誌傳入該命令。

```
leaf8# vsh_lc -c "show platform internal hal health-stats"
|Sandbox_ID: 0  Asic Bitmap: 0x0
|-------------------------------------
 ...
Policy stats:
=============
policy_count                  : 96
max_policy_count              : 65536
policy_otcam_count            : 175
max_policy_otcam_count            : 8192
policy_label_count                : 0
max_policy_label_count            : 0
=============
```

EPG到EPG

常規策略丟棄注意事項

有多種方法可以解決兩個端點之間的連線問題。以下方法為快速有效地隔離連線問題是否是策略丟棄（合約誘導）的結果提**供了**良好的起點。

一些在深入之前值得詢問的高級問題：

- 終端是否處於相同或不同的EPG中？ 位於不同EPG(Inter-EPG)中的兩個端點之間的流量會遭到隱式拒絕，因此需要聯絡以允許通訊。除非使用的是EPG內隔離，否則隱式允許同一EPG內的兩個端點之間的流量（EPG內）。
- VRF是強制還是不強制執行？ 當VRF處於**強制**模式時（在VRF中），需要合約才能使兩個不同

EPG中的終端進行通訊。當VRF處於**未實施**模式時（在VRF內），ACI交換矩陣將允許所有流量跨屬於未實施VRF的多個EPG，無論應用哪個ACI合約。

方法

藉助各種工具，有些工具比其他工具更合適和更方便起步，這取決於已經瞭解的有關受影響流量的資訊級別。

ACI交換矩陣中資料包的完整路徑是否為已知（入口枝葉、出口枝葉……）？

- 如果答案為是，則應使用ELAM Assistant來識別源或目標交換機上的丟棄原因。
- 如果答案為否，Visibility & Troubleshooting、fTriage、contract_parser、Tenant檢視中的Operational頁籤和iBash命令將有助於縮小資料包的路徑或使丟棄原因更清晰。

請注意，本部分不會詳細討論fTriage工具。有關使用此工具的更多詳細資訊，請參閱「交換矩陣內轉發」一章。

請考慮，雖然可視性和故障排除有助於快速直觀地顯示兩個端點之間的資料包丟棄位置，但fTriage顯示更多深入資訊，以便進行進一步的故障排除。即fTriage將幫助識別介面、丟棄原因以及其他有關受影響流的低級詳細資訊

.

此示例場景將展示如何對兩個端點之間的策略丟棄進行故障排除：192.168.21.11和192.168.23.11

假設這兩個端點之間存在丟包情況，將使用以下故障排除工作流程確定問題的根本原因：

確定流量中涉及的src/dst枝葉：

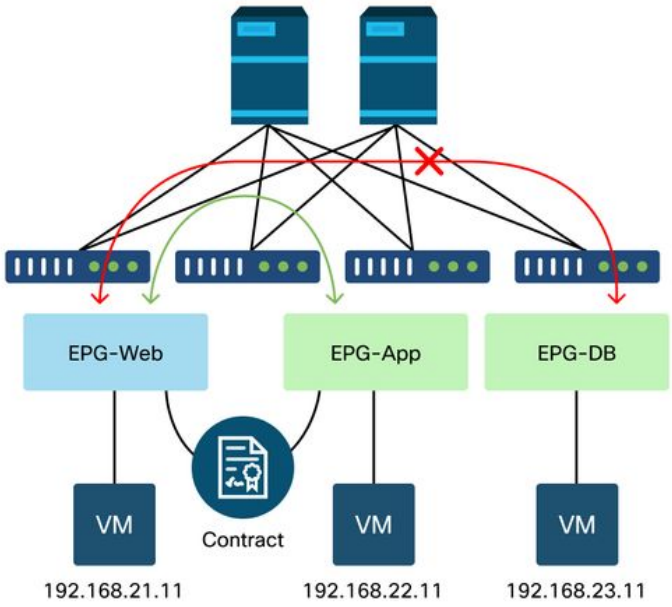
1. 使用**可見性和故障排除**跟蹤資料包流並確定哪個裝置正在丟棄資料包。
2. 在所選裝置上運行命令show logging ip access-list internal packet-log deny。 如果拒絕並記錄具有其中一個相關IP地址的資料包，則**packet-log**將按點選次數列印相關終端和約定名稱。
3. 在源枝葉和目標枝葉上使用命令「contract_parser.py —vrf <tenant>:<VRF>」來觀察已配置合約的命中計數： 如果封包在來源交換器或目的地交換器上達成協定，相關協定的計數器會增加在許多流可能達到相同規則（兩個相關的EPG之間的許多端點/流）的情況下，此方法比IP訪問清單內部資料包日誌的粒度更小。

上述步驟將在下一段進一步說明。

故障排除場景EPG到EPG的示例

此示例場景將展示如何對兩個端點之間的策略丟棄進行故障排除：EPG-Web中的192.168.21.11和EPG-DB中的192.168.23.11。

拓撲

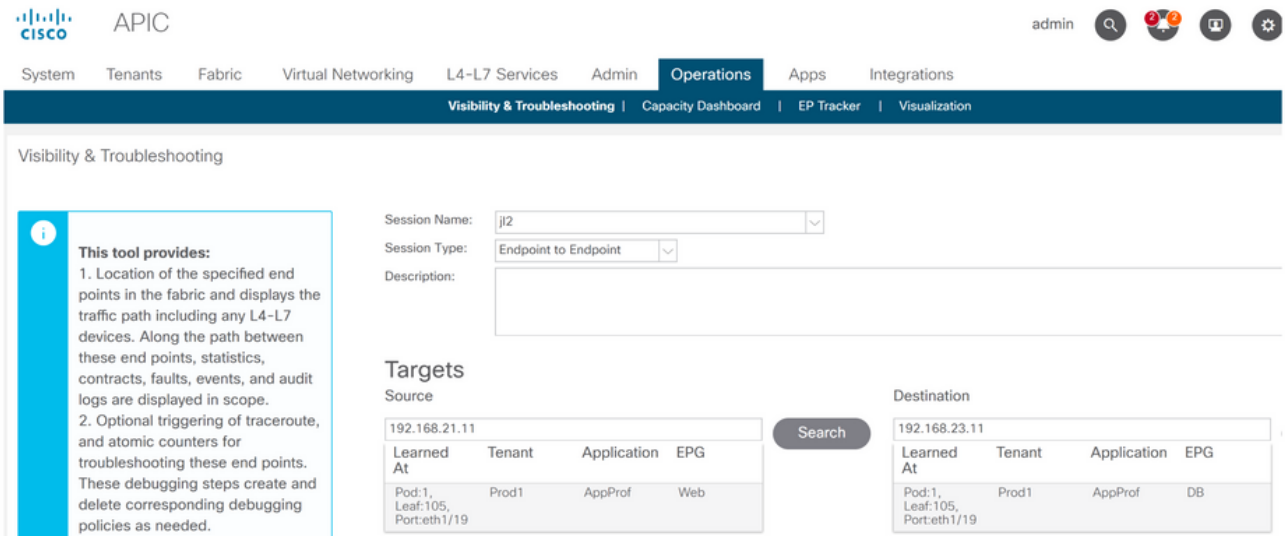


確定資料包丟棄中涉及的源枝葉交換機和目標枝葉交換機

可視性與故障排除

可視性和故障排除工具將幫助視覺化特定EP到EP流發生資料包丟棄的交換機，並識別可能丟棄資料包的位置。

可視性與故障排除的配置

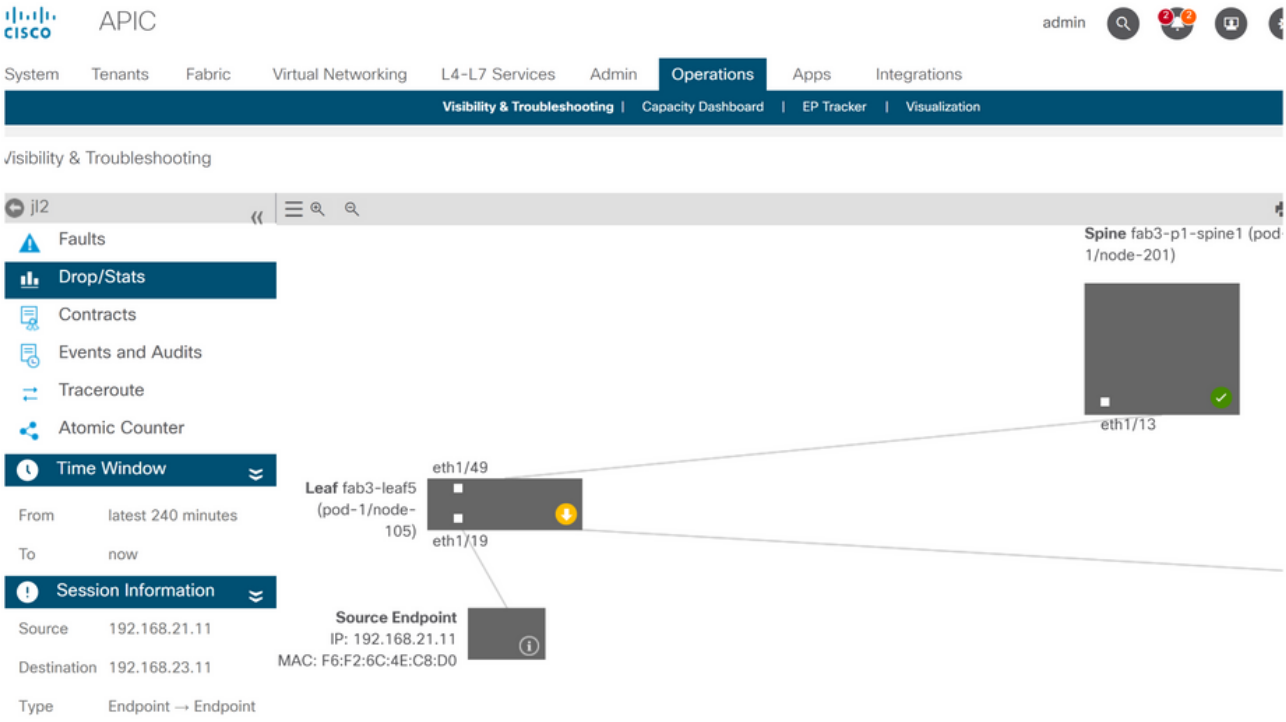


配置會話名稱、源和目標終結點。然後按一下「提交」或「生成報告」。

該工具將自動在交換矩陣中查詢終端，並提供有關EP所屬的租戶、應用配置檔案和EPG的資訊。

在這種情況下，將發現EP屬於租戶Prod1，它們屬於同一個應用程式配置檔案「AppProf」，並被分配到不同的EPG:「Web」和「DB」。

丟棄標識

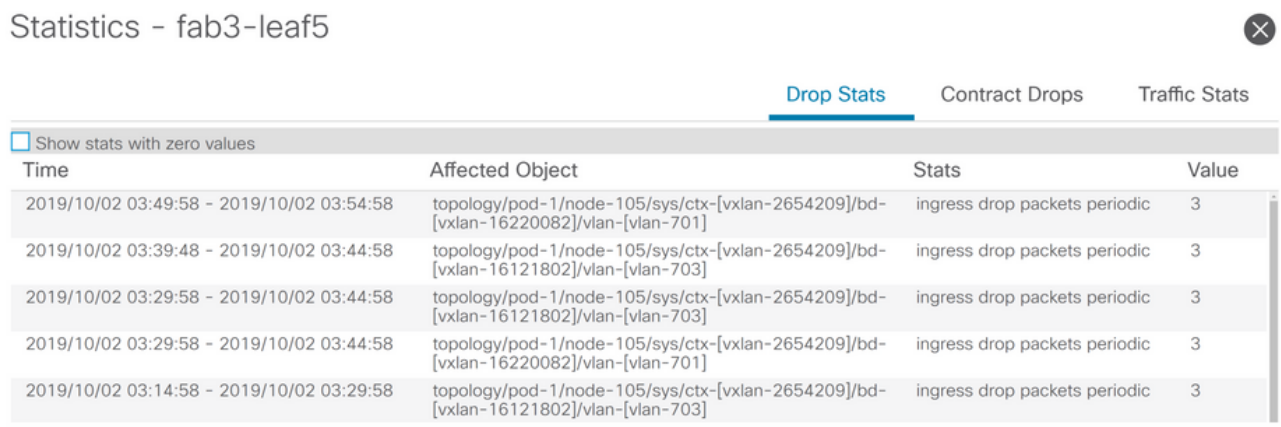


該工具將自動呈現故障排除方案的拓撲。在這種情況下，兩個端點恰好連線到同一個枝葉交換機。

通過導航到Drop/Stats子選單，使用者可以檢視有關枝葉或脊柱上的常規丟包。請參閱本書「交換矩陣內轉發」一章中的「介面丟棄」部分，瞭解更多有關瞭解哪些丟棄相關的資訊。

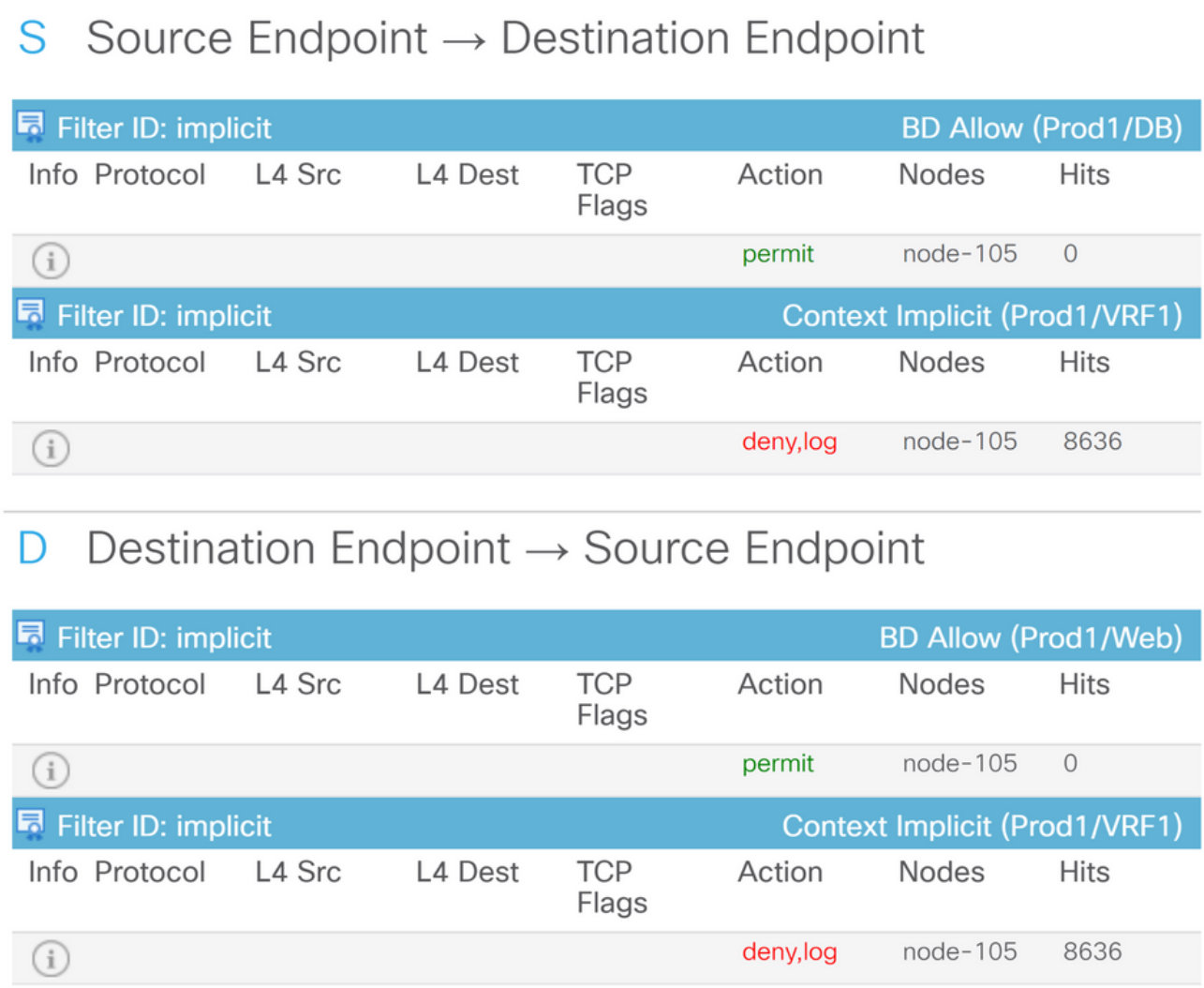
這些丟棄中有許多是預期行為，可以忽略。

刪除詳細資訊



使用交換機圖上的黃色「Packets dropped」按鈕向下鑽取以丟棄詳細資訊，使用者可以檢視有關丟棄的流的詳細資訊。

合約詳細資訊



通過導航到Contracts子選單，使用者可確定哪份合約導致EPG之間的策略離線。在本例中，Implicit to Deny Prod1/VRF1顯示一些命中。這並不一定表示指定的流（192.168.21.11和192.168.23.11）正在達到此隱含的deny。如果上下文隱式拒絕規則的命中數增加，則意味著Prod1/DB和Prod1/Web之間存在未命中任何合約的流量，因此隱式拒絕將丟棄該流量。

在Tenant（租戶）>的Application Profile Topology（應用程式配置檔案拓撲）檢視中，選擇左邊的Application Profile name（應用程式配置檔名稱）> Topology（拓撲），可以驗證哪些合約應用到了資料庫EPG。在這種情況下，不會將合約分配給EPG:

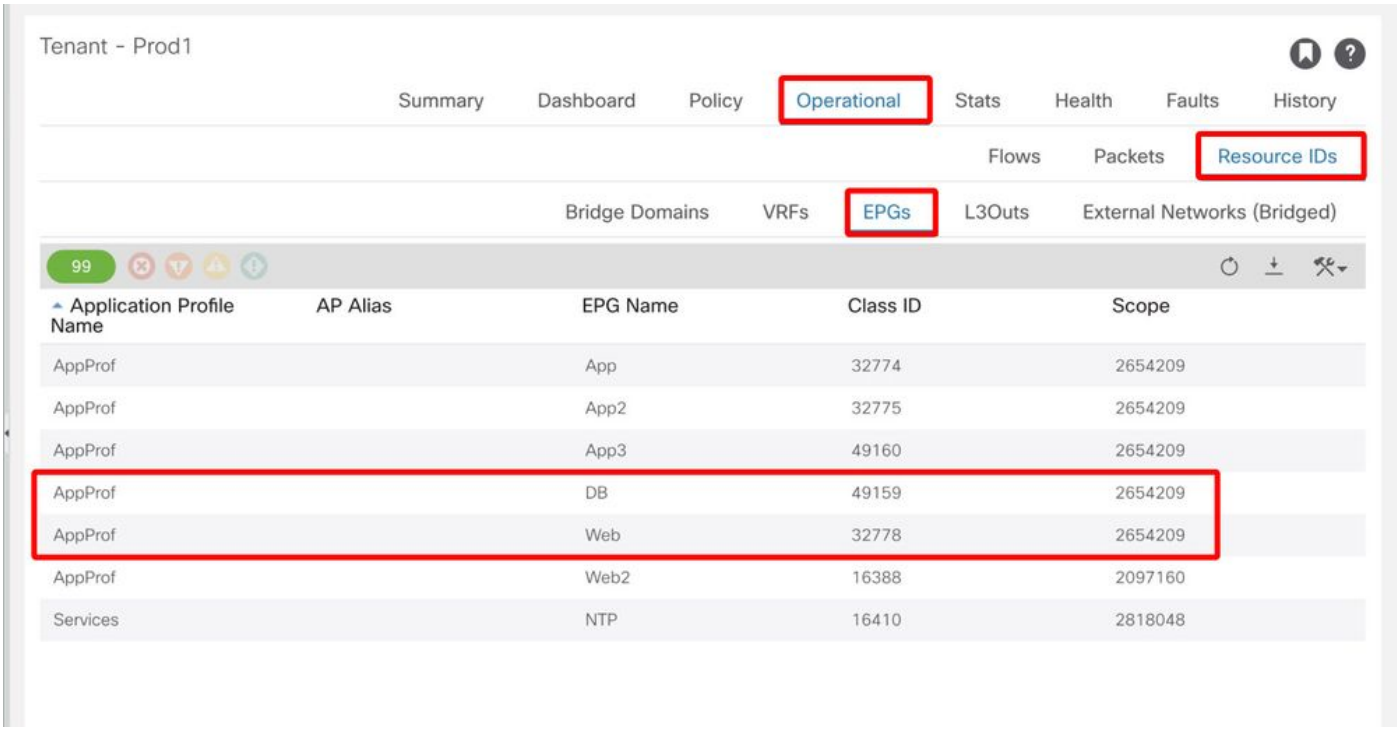
合約視覺化

既然源EPG和目標EPG已知，還可以確定其他相關資訊，例如：

- 受影響終端的src/dst **EPG pcTag**。pcTag是用於使用分割槽規則標識EPG的類ID。
- 受影響端點的src/dst **VRFVNID**，也稱為**scope**。

通過開啟租戶>選擇左側租戶名稱>操作>資源ID > EPG，可以從APIC GUI輕鬆檢索類ID和範圍

用於查詢EPG pcTag和範圍的租戶資源ID



在本例中，類ID和範圍是：

- Web EPG pcTag模32778
- Web EPG作用域2654209
- DB EPG pcTag 49159
- DB EPG作用域2654209

驗證應用於正在故障排除的流量的策略

iBash

驗證ACI枝葉上丟棄的資料包的有趣工具是iBash命令列：'show logging ip access-list internal packet-log deny':

```
leaf5# show logging ip access-list internal packet-log deny | grep 192.168.21.11
[2019-10-01T14:25:44.746528000+09:00]: CName: Prod1:VRF1(VXLAN: 2654209), VlanType: FD_VLAN,
Vlan-Id: 114, SMac: 0xf6f26c4ec8d0, DMac:0x0022bdf819ff, SIP: 192.168.21.11, DIP: 192.168.23.11,
SPort: 0, DPort: 0, Src Intf: Ethernet1/19, Proto: 1, PktLen: 126
[2019-10-01T14:25:44.288653000+09:00]: CName: Prod1:VRF1(VXLAN: 2654209), VlanType: FD_VLAN,
Vlan-Id: 116, SMac: 0x3e2593f0eded, DMac:0x0022bdf819ff, SIP: 192.168.23.11, DIP: 192.168.21.11,
SPort: 0, DPort: 0, Src Intf: Ethernet1/19, Proto: 1, PktLen: 126
```

根據前面的輸出，可以看到，在枝葉交換器上，來源為EP 192.168.23.11到192.168.21.11的許多ICMP封包已捨棄。

contract_parser工具將幫助驗證應用於終端所關聯的VRF的實際策略：

```
leaf5# contract_parser.py --vrf Prod1:VRF1
Key:
[prio:RuleId] [vrf:{str}] action protocol src-epg [src-l4] dst-epg [dst-l4]
[flags][contract:{str}] [hit=count]

[7:5159] [vrf:Prod1:VRF1] permit ip tcp tn-Prod1/ap-App1/epg-App(32771) eq 5000 tn-Prod1/ap-
```

```
App1/epg-Web(32772)  [contract:uni/tn-Prod1/brc-web_to_app] [hit=0]
[7:5156] [vrf:Prod1:VRF1] permit ip tcp tn-Prod1/ap-App1/epg-Web(32772) tn-Prod1/ap-App1/epg-
App(32771) eq 5000  [contract:uni/tn-Prod1/brc-web_to_app] [hit=0]
[16:5152] [vrf:Prod1:VRF1] permit any epg:any tn-Prod1/bd-Web(49154) [contract:implicit] [hit=0]
[16:5154] [vrf:Prod1:VRF1] permit arp epg:any epg:any [contract:implicit] [hit=0]
[21:5155] [vrf:Prod1:VRF1] deny,log any epg:any epg:any [contract:implicit] [hit=38,+10]
[22:5153] [vrf:Prod1:VRF1] deny,log any epg:any pfx-0.0.0.0/0(15) [contract:implicit] [hit=0]
```

這也可以通過在枝葉中程式設計的分割槽規則驗證交換機實施的策略。

```
leaf5# show zoning-rule scope 2654209
+---------+--------+--------+----------+----------------+---------+----------+------------+-----
-----+-------------------------+
| Rule ID | SrcEPG | DstEPG | FilterID |      Dir       |  operSt |  Scope   |    Name    |
Action  |         Priority        |
+---------+--------+--------+----------+----------------+---------+----------+------------+-----
-----+-------------------------+
|   5155  |   0    |   0    | implicit |    uni-dir     | enabled | 2654209  |            |
deny,log |     any_any_any(21)     |
|   5159  | 32771  | 32772  |   411    | uni-dir-ignore | enabled | 2654209  | web_to_app |
permit  |      fully_qual(7)      |
|   5156  | 32772  | 32771  |   410    |     bi-dir     | enabled | 2654209  | web_to_app |
permit  |      fully_qual(7)      |
+---------+--------+--------+----------+----------------+---------+----------+------------+-----
-----+-------------------------+
```

如可視性和故障排除工具、contract_parser工具和分割槽規則所示，輸出確認在故障排除過程中，源和目標EPG之間沒有合約。很容易假設丟棄的資料包與隱式拒絕規則5155匹配。

ELAM捕獲

ELAM捕獲提供用於檢查轉發詳細資訊的ASIC級別報告，在丟包的情況下，該報告指示丟棄原因。如果刪除的原因是策略刪除（如本場景所示），則ELAM捕獲的輸出將如下所示。

請注意，本章將不會討論有關設定ELAM捕獲的詳細資訊，請參閱「交換矩陣內轉發」一章。

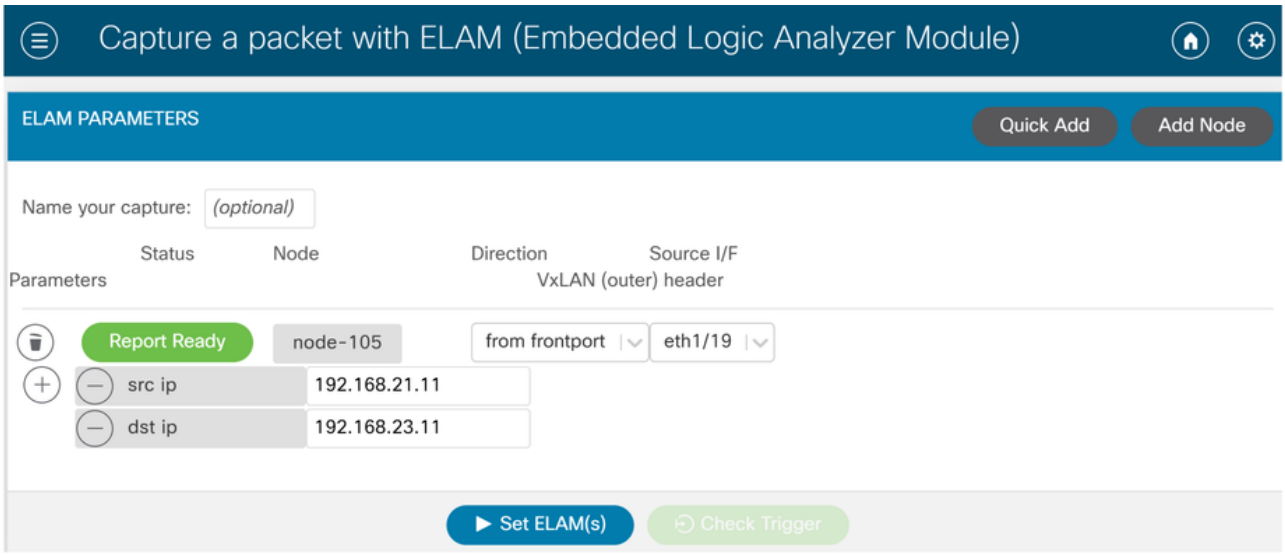
```
leaf5# vsh_lc
module-1# debug platform internal tah elam asic 0
module-1(DBG-elam)# trigger init in-select 6 out-select 0
module-1(DBG-elam)# trigger reset
module-1(DBG-elam-insel6)# set outer ipv4 src_ip 192.168.21.11 dst_ip 192.168.23.11
module-1(DBG-elam-insel6)# start
module-1(DBG-elam-insel6)# status

ELAM STATUS
===========
Asic 0 Slice 0 Status Triggered
Asic 0 Slice 1 Status Armed

module-1(DBG-elam-insel6)# ereport | grep reason
RW drop reason                          : SECURITY_GROUP_DENY
LU drop reason                          : SECURITY_GROUP_DENY
   pkt.lu_drop_reason: 0x2D
```

上面的ELAM報告清楚地顯示資料包由於策略丟棄而被丟棄：'SECURITY_GROUP_DENY'

ELAM助理：

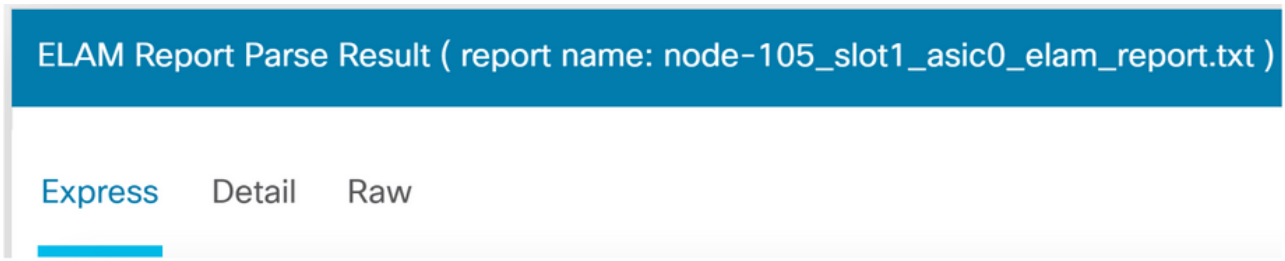
通過APIC GUI上的ELAM Assistant應用可以顯示ELAM捕獲的相同結果。

組態



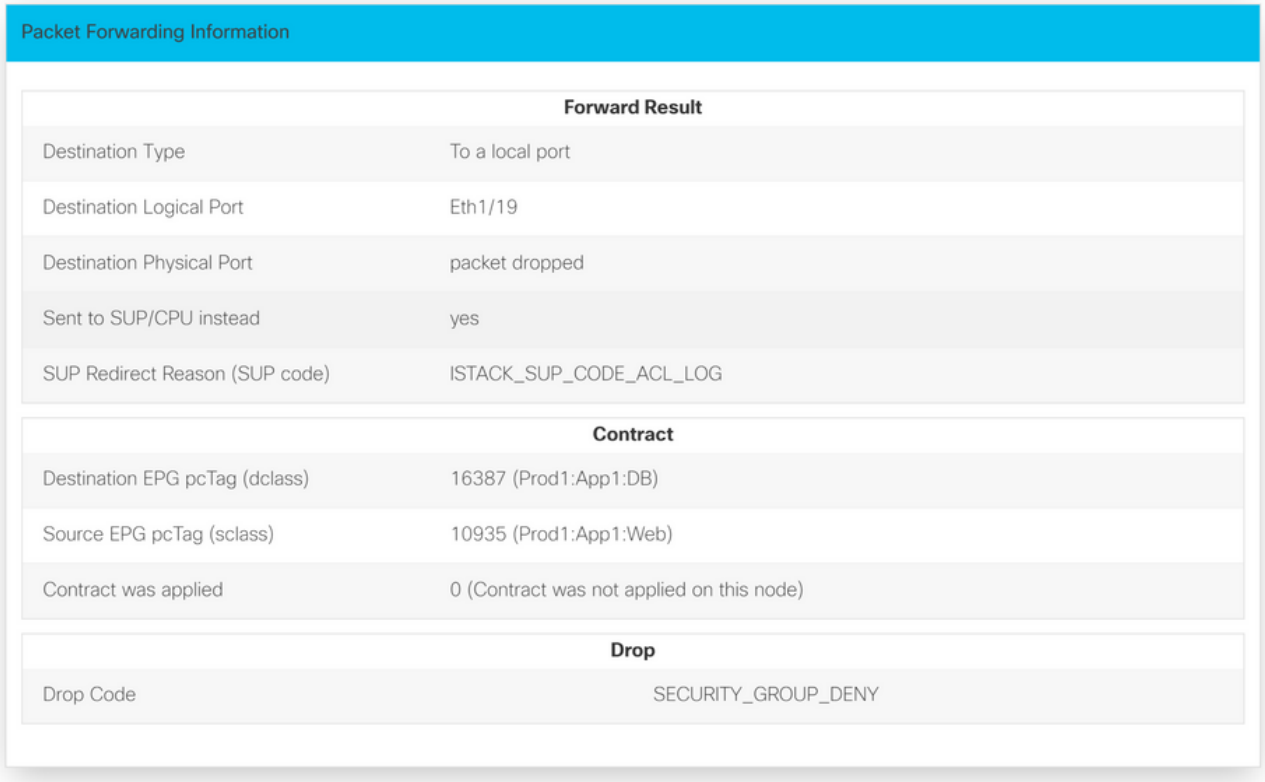
通常，使用者會為興趣流配置源和目標詳細資訊。在此示例中，源IP用於捕獲目標EPG中指向終結點的流量，該終結點與源EPG沒有合約關係。

Elam Assistant Express報告



使用ELAM Assistant可以檢視三個級別的輸出。這些是Express、Detail和Raw。

Elam Assistant Express報告（續）



在Express Result（快速結果）下，丟棄代碼原因SECURITY_GROUP_DENY表示丟棄是合約命中的結果。

首選組

關於合約首選組

在配置了合約首選組的VRF中，有兩種型別的策略實施可用於EPG:

- 包括的EPG:EPG可以在沒有合約的情況下自由地相互通訊，如果他們擁有合約首選組的成員資格。這基於source-any-destination-any-permit預設規則。
- 排除的EPG:非首選組成員的EPG需要合約才能相互通訊。否則，將應用排除EPG和任何EPG之間的拒絕規則。

合約首選組功能能夠更好地控制VRF中EPG之間的通訊。如果VRF中的大部分EPG應該具有開放式通訊，但少數的EPG應該只具有與其他EPG的有限通訊，請配置合約首選組和帶有過濾器的合約的組合，以便更精確地控制EPG間通訊。

從首選組排除的EPG只能與其他EPG通訊，前提是存在覆蓋source-any-destination-any-deny預設規則的合約。

合約首選組程式設計

從本質上講，合約首選組與常規合約相反。對於常規合約，顯式permit zoning-rules使用帶VRF Scope的隱式deny zoning-rule進行程式設計。對於首選組，隱式的PERMIT分割槽規則是使用最高數值優先順序值程式設計的，而特定的DENY分割槽規則則程式設計為禁止來自非首選組成員的EPG的流量。因此，首先評估拒絕規則，如果流與這些規則不匹配，則隱式允許流。

對於首選組之外的每個EPG，始終有一對明確的deny zoning-rules:

- 一個從非首選組成員到任何pcTag（值0）。
- 另一個從任何pcTag（值0）到非首選組成員。

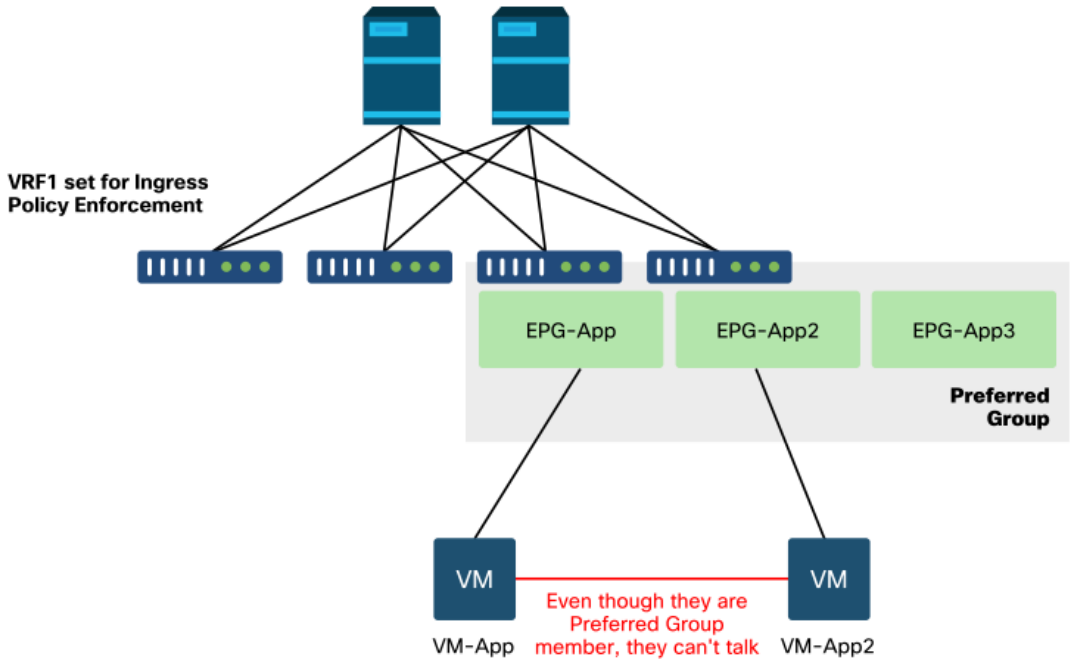
首選組故障排除場景

下圖顯示了一個邏輯拓撲，其中EPG應用、App2和App3均配置為首選組成員。

VM-App是EPG-App的一部分，VM-App2是EPG-App2的一部分。App和App2 EPG都應是首選的一部分，因此應自由通訊。

VM-App在TCP埠6000上向VM-App2發起流量流。EPG-App和EPG-App2都是VRF1的首選組成員。VM-App2從TCP埠6000上從不接收任何資料包。

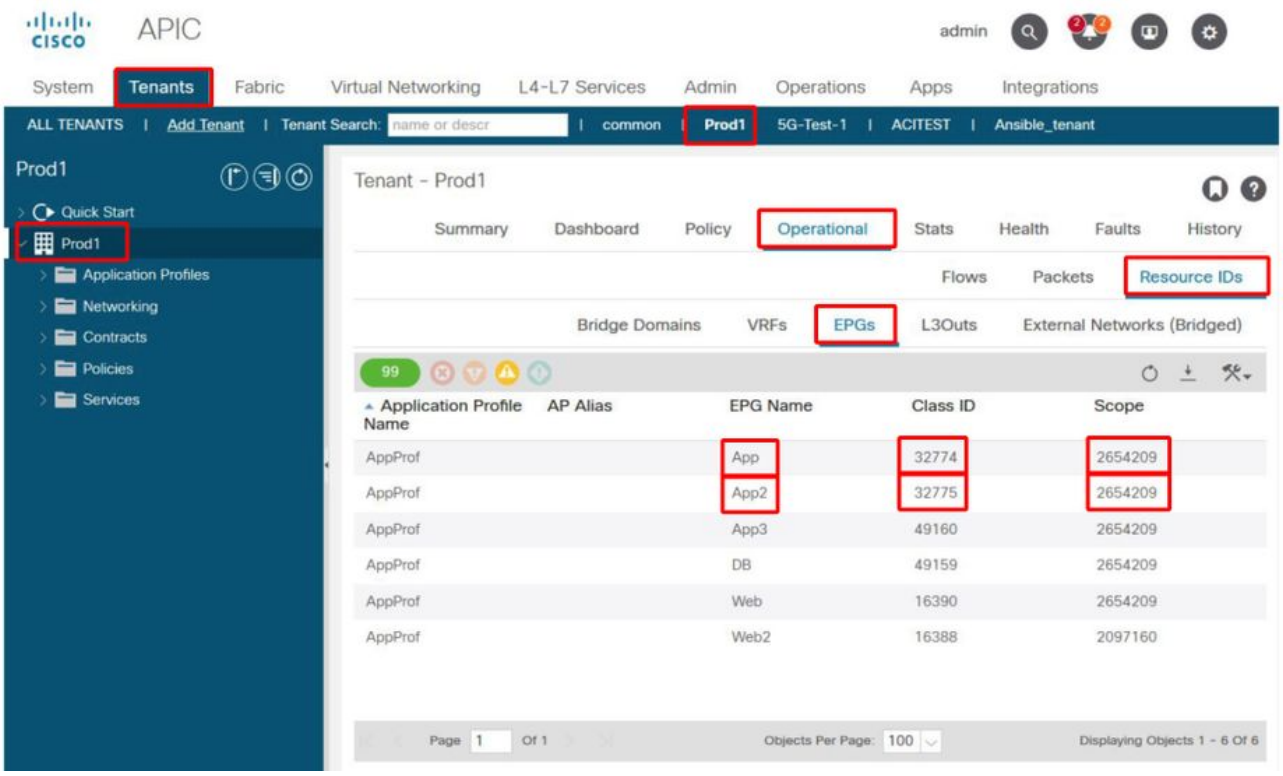
拓撲



工作流程

1.查詢EPG APP的pcTag及其VRF VNID/Scope

EPG和VRF pcTags



2.使用入口枝葉上的contract_parser.py驗證合約程式設計

使用contract_parser.py和/或「show zoning-rule」命令並指定VRF

```
fab3-leaf8# show zoning-rule scope 2654209
+---------+--------+--------+----------+---------+---------+---------+------+----------+--------
--------------------+
| Rule ID | SrcEPG | DstEPG | FilterID |   Dir   |  operSt |  Scope  | Name |  Action
|          Priority          |
+---------+--------+--------+----------+---------+---------+---------+------+----------+--------
--------------------+
|   4165  |   0    |   0    | implicit | uni-dir | enabled | 2654209 |      |  permit  |
grp_any_any_any_permit(20) |
|   4160  |   0    |   0    | implarp  | uni-dir | enabled | 2654209 |      |  permit  |
any_any_filter(17)     |
|   4164  |   0    |   15   | implicit | uni-dir | enabled | 2654209 |      | deny,log |
grp_any_dest_any_deny(19)  |
|   4176  |   0    | 16386  | implicit | uni-dir | enabled | 2654209 |      |  permit  |
any_dest_any(16)       |
|   4130  | 32770  |   0    | implicit | uni-dir | enabled | 2654209 |      | deny,log |
grp_src_any_any_deny(18)   |
|   4175  | 49159  |   0    | implicit | uni-dir | enabled | 2654209 |      | deny,log |
grp_src_any_any_deny(18)   |
|   4129  |   0    | 49159  | implicit | uni-dir | enabled | 2654209 |      | deny,log |
grp_any_dest_any_deny(19)  |
|   4177  | 32778  |   0    | implicit | uni-dir | enabled | 2654209 |      | deny,log |
grp_src_any_any_deny(18)   |
|   4128  |   0    | 32778  | implicit | uni-dir | enabled | 2654209 |      | deny,log |
grp_any_dest_any_deny(19)  |
|   4178  | 32775  |   0    | implicit | uni-dir | enabled | 2654209 |      | deny,log |
grp_src_any_any_deny(18)   |
|   4179  |   0    | 32775  | implicit | uni-dir | enabled | 2654209 |      | deny,log |
grp_any_dest_any_deny(19)  |
+---------+--------+--------+----------+---------+---------+---------+------+----------+--------
--------------------+

fab3-leaf8# contract_parser.py --vrf Prod1:VRF1
```

```
Key:
[prio:RuleId] [vrf:{str}] action protocol src-epg [src-l4] dst-epg [dst-l4]
[flags][contract:{str}] [hit=count]
[16:4176] [vrf:Prod1:VRF1] permit any epg:any tn-Prod1/bd-App(16386) [contract:implicit] [hit=0]
[16:4160] [vrf:Prod1:VRF1] permit arp epg:any epg:any [contract:implicit] [hit=0]
[18:4130] [vrf:Prod1:VRF1] deny,log any tn-Prod1/vrf-VRF1(32770) epg:any [contract:implicit]
[hit=?]
[18:4178] [vrf:Prod1:VRF1] deny,log any epg:32775 epg:any [contract:implicit] [hit=?]
[18:4177] [vrf:Prod1:VRF1] deny,log any epg:32778 epg:any [contract:implicit] [hit=?]
[18:4175] [vrf:Prod1:VRF1] deny,log any epg:49159 epg:any [contract:implicit] [hit=?]
[19:4164] [vrf:Prod1:VRF1] deny,log any epg:any pfx-0.0.0.0/0(15) [contract:implicit] [hit=0]
[19:4179] [vrf:Prod1:VRF1] deny,log any epg:any epg:32775 [contract:implicit] [hit=?]
[19:4128] [vrf:Prod1:VRF1] deny,log any epg:any epg:32778 [contract:implicit] [hit=?]
[19:4129] [vrf:Prod1:VRF1] deny,log any epg:any epg:49159 [contract:implicit] [hit=?]
[20:4165] [vrf:Prod1:VRF1] permit any epg:any epg:any [contract:implicit] [hit=65]
```

檢查上述輸出，觀察到具有最高優先順序20的隱式permit entry — ruleId 4165。除非有優先順序較低的顯式拒絕規則禁止流量，否則此隱式允許規則將導致允許所有流量。

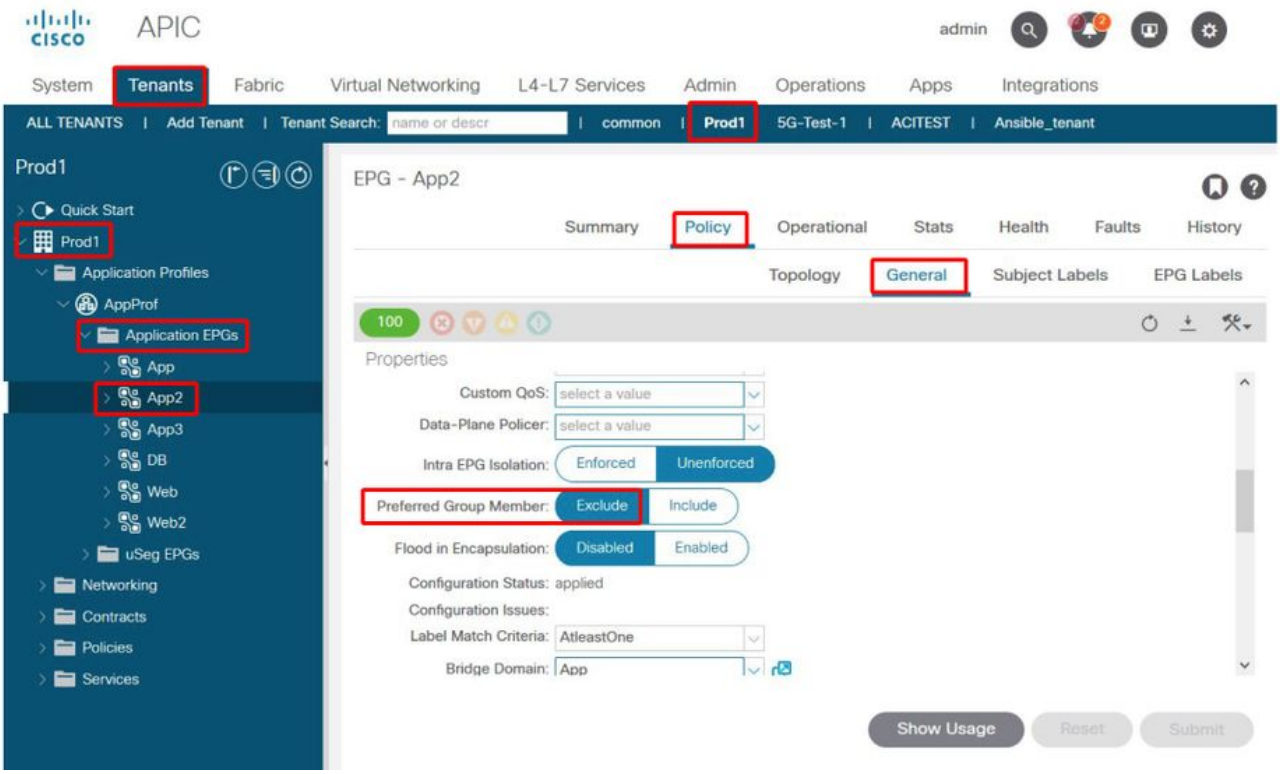
此外，針對pcTag 32775（即EPG App2的pcTag）觀察到兩個顯式拒絕規則。這兩個顯式拒絕分割槽規則禁止從任何EPG到EPG App2的流量，反之亦然。這些規則的優先順序為18和19，因此它們將優先於預設允許規則。

結論是EPG App2不是首選組成員，因為已遵守顯式拒絕規則。

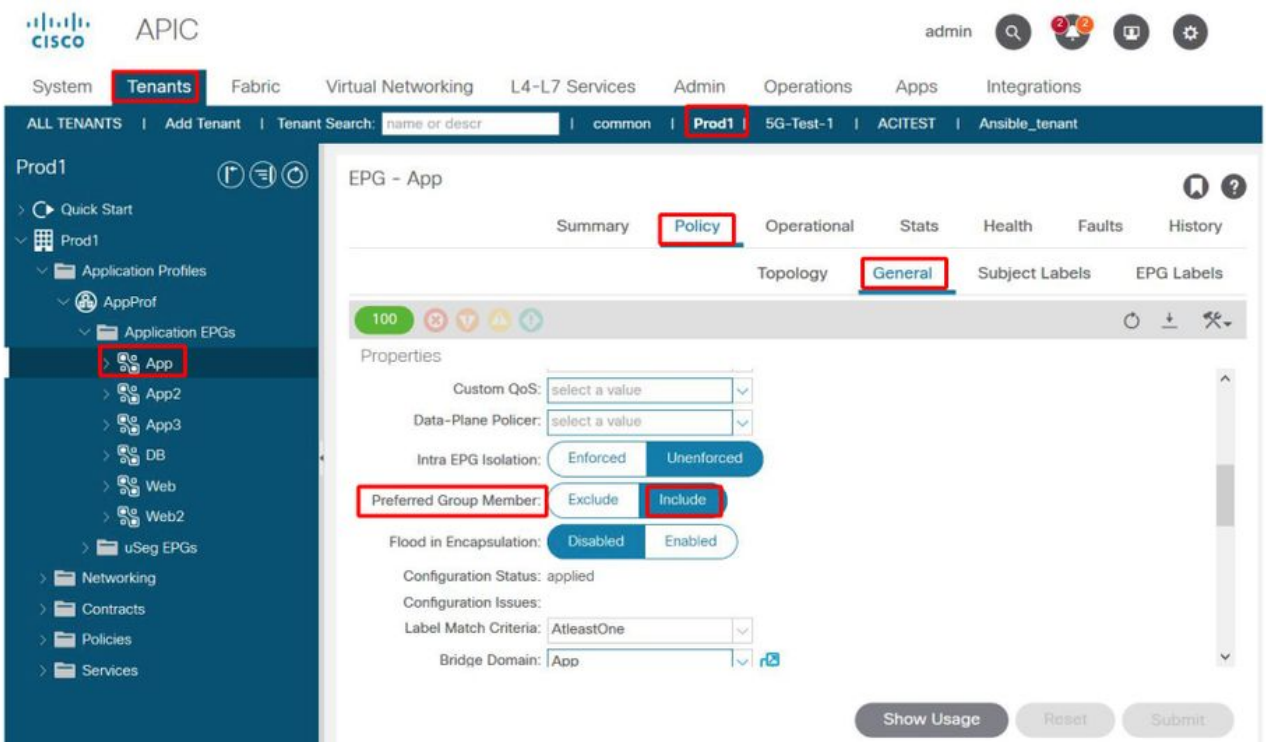
3.檢驗EPG首選組成員配置

導航APIC GUI並檢查EPG App2和EPG應用首選組成員配置，在下圖中，請參閱EPG App2未配置為首選組成員。

EPG App2 — 排除首選組成員設定



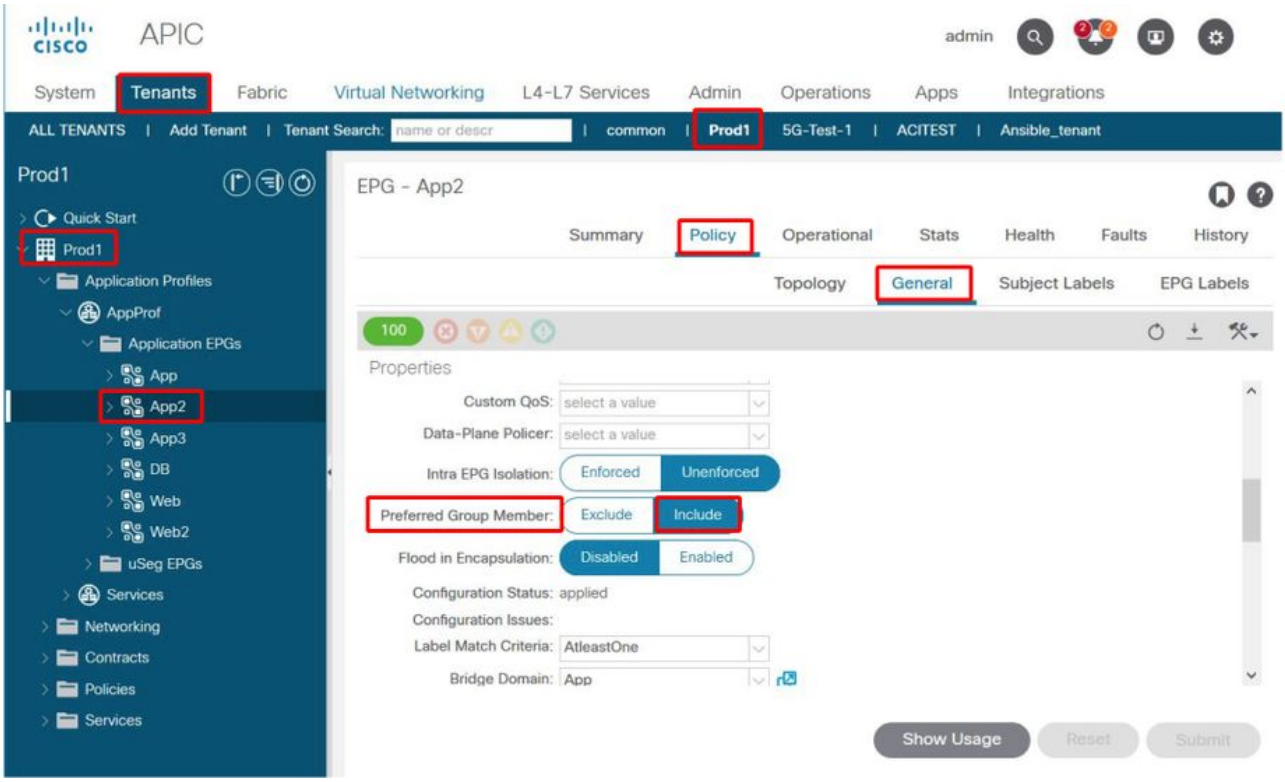
EPG應用 — 包括首選組成員設定



4.將EPG App2設定為首選組成員

更改App2 EPG配置可使首選組作為首選組的一部分自由通訊。

EPG App2 — 包括首選組成員設定



5.使用src EP所在的枝葉上的contract_parser.py重新驗證合約程式設計

再次使用contract_parser.py並指定VRF名稱以驗證EPG App2的顯式拒絕規則是否已消失。

```
fab3-leaf8# contract_parser.py --vrf Prod1:VRF1
```

```
Key:
[prio:RuleId] [vrf:{str}] action protocol src-epg [src-l4] dst-epg [dst-l4]
[flags][contract:{str}] [hit=count]
[16:4176] [vrf:Prod1:VRF1] permit any epg:any tn-Prod1/bd-App(16386) [contract:implicit] [hit=0]
[16:4160] [vrf:Prod1:VRF1] permit arp epg:any epg:any [contract:implicit] [hit=0]
[18:4175] [vrf:Prod1:VRF1] deny,log any epg:16390 epg:any [contract:implicit] [hit=0]
[18:4167] [vrf:Prod1:VRF1] deny,log any epg:23 epg:any [contract:implicit] [hit=0]
[18:4156] [vrf:Prod1:VRF1] deny,log any tn-Prod1/vrf-VRF1(32770) epg:any [contract:implicit]
[hit=0]
[18:4168] [vrf:Prod1:VRF1] deny,log any epg:49159 epg:any [contract:implicit] [hit=0]
[19:4164] [vrf:Prod1:VRF1] deny,log any epg:any pfx-0.0.0.0/0(15) [contract:implicit] [hit=0]
[19:4169] [vrf:Prod1:VRF1] deny,log any epg:any epg:16390 [contract:implicit] [hit=0]
[19:4159] [vrf:Prod1:VRF1] deny,log any epg:any epg:23 [contract:implicit] [hit=0]
[19:4174] [vrf:Prod1:VRF1] deny,log any epg:any epg:49159 [contract:implicit] [hit=0]
[20:4165] [vrf:Prod1:VRF1] permit any epg:any epg:any [contract:implicit] [hit=65]
```

在上面的輸出中不再觀察到EPG App2及其pcTag 32775的顯式拒絕規則。這意味著EPG應用中的EP與EPG應用2之間的流量現在將匹配隱含的允許規則(ruleId 4165)，優先順序最高為20。

vzAny到EPG

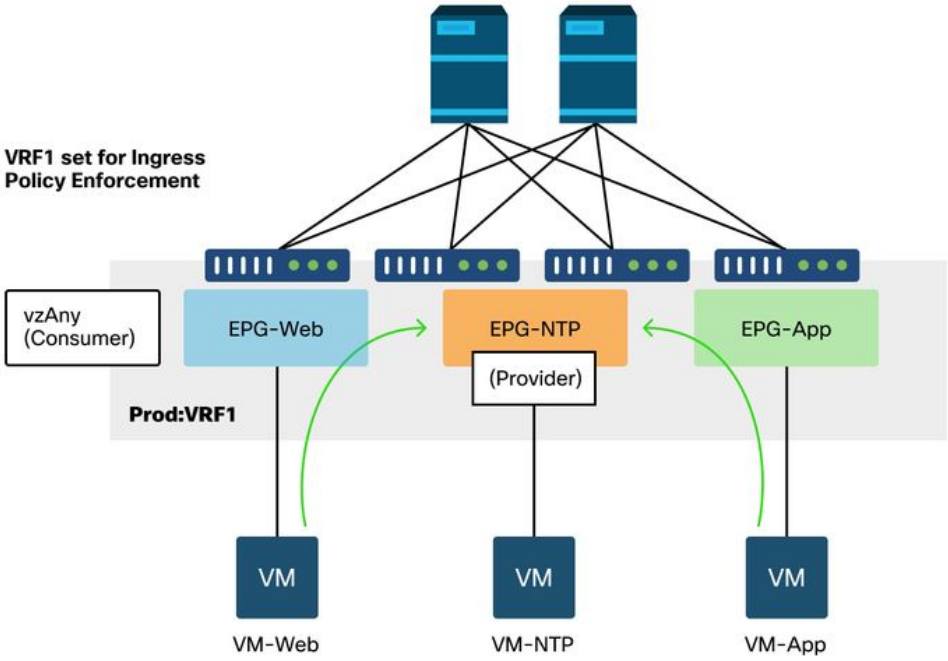
關於vzAny

在一個或多個EPG之間配置合約時，可以將合約配置為消耗關係或提供的關係。當EPG數量增加時，它們之間的合約關係量也會增加。某些常見用例要求所有EPG與另一個特定EPG交換流量。這種用例可以是包含EP的EPG，該EP提供需要由相同VRF（例如NTP或DNS）內的所有其他EPG消費的服務。vzAny在配置所有EPG與提供由所有其他EPG使用的服務的特定EPG之間的合約關係時允許降低運營開銷。此外，由於每個vzAny合約關係僅新增了2個分割槽規則，因此vzAny允許在枝葉交換機上更高效地使用安全策略CAM。

用例示例

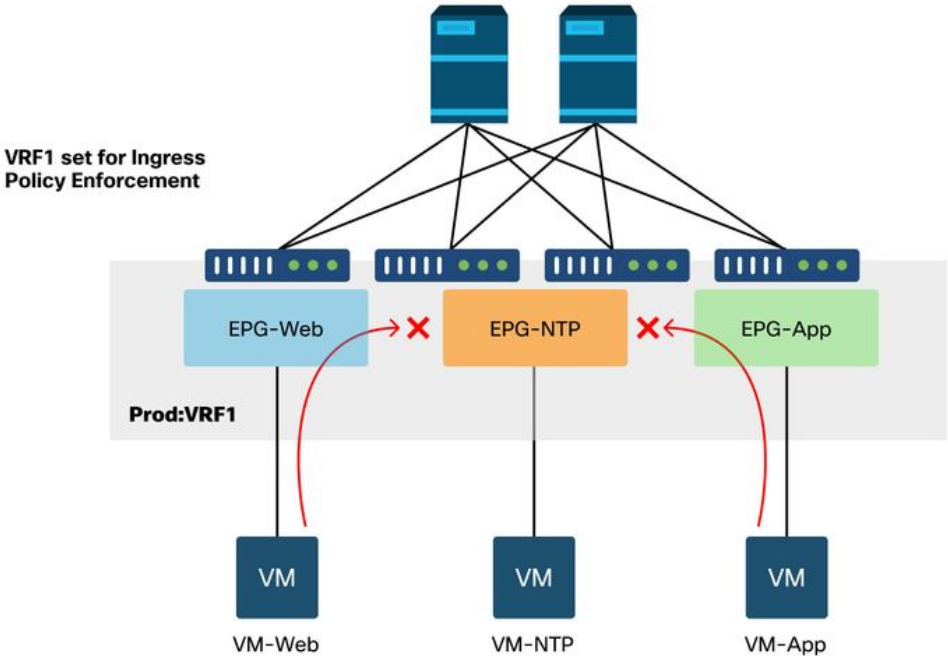
下圖描述了這樣的使用案例，即EPG的Web和App中的VM-Web和VM-App分別需要從EPG-NTP中的VM-NTP使用NTP服務。vzAny不在EPG NTP上配置提供的合約，並且隨後具有與EPG Web和App上使用的合約相同的合約，而是允許VRF產品中的每個EPG使用EPG NTP的NTP服務。

vzAny - VRF中的任何EPG Prod:VRF1都可以使用EPG NTP中的NTP服務



考慮以下情況：當使用NTP服務的EPG之間沒有合約時，會觀察到丟棄。

疑難排解案例 — 如果沒有合約，流量會捨棄

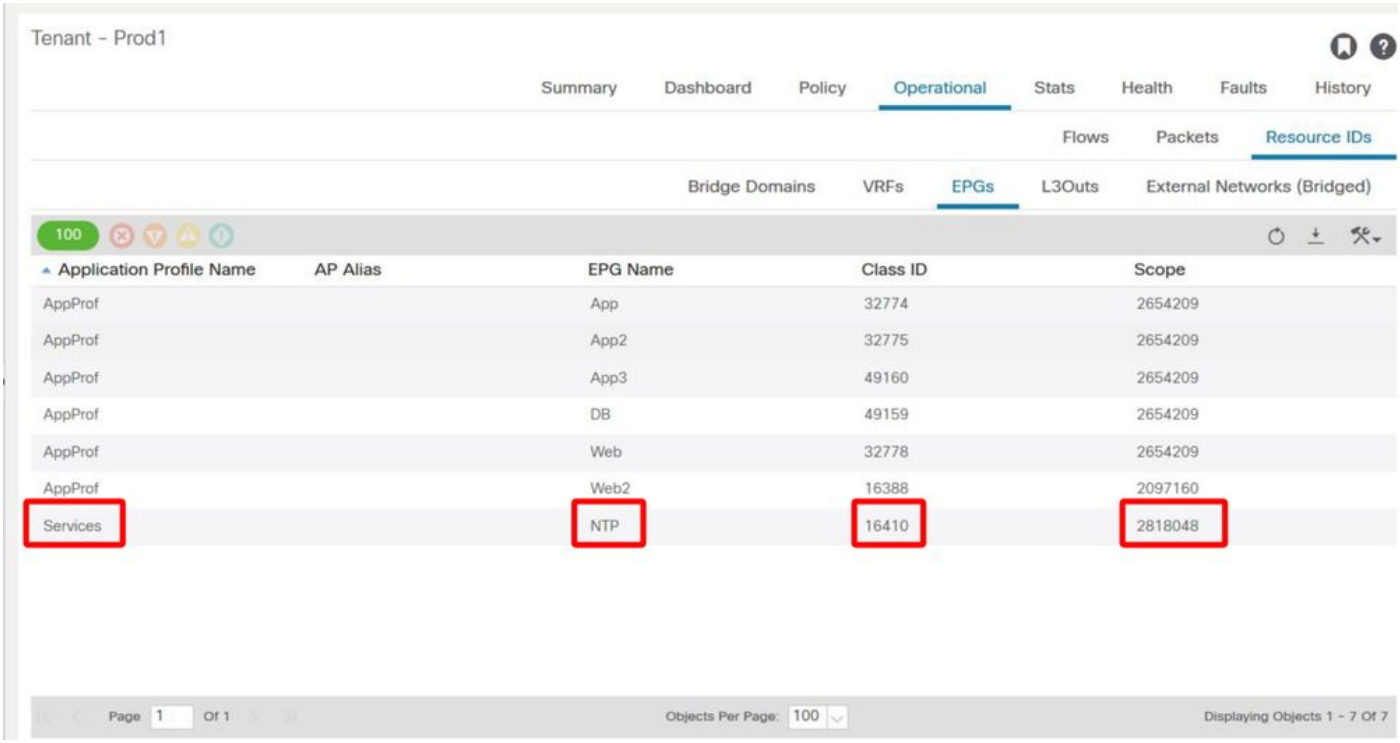


工作流程

1.查詢EPG NTP的pcTag及其VRF VNID/Scope

「Tenant > Operational > Resource IDs > EPGs」允許查詢pcTag和範圍

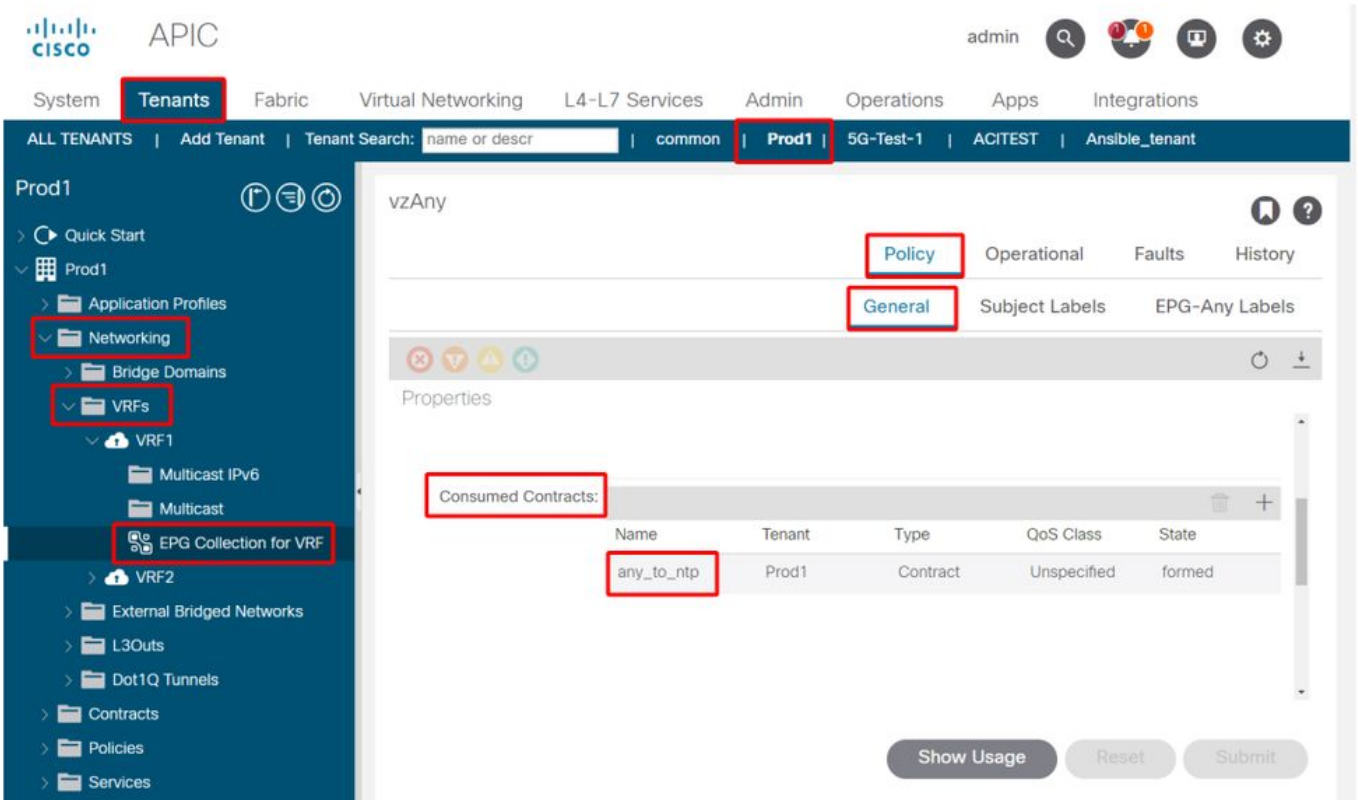
EPG NTP pcTag及其VRF VNID/範圍



2.驗證是否已將合約配置為vz作為VRF的一部分使用的任何合約

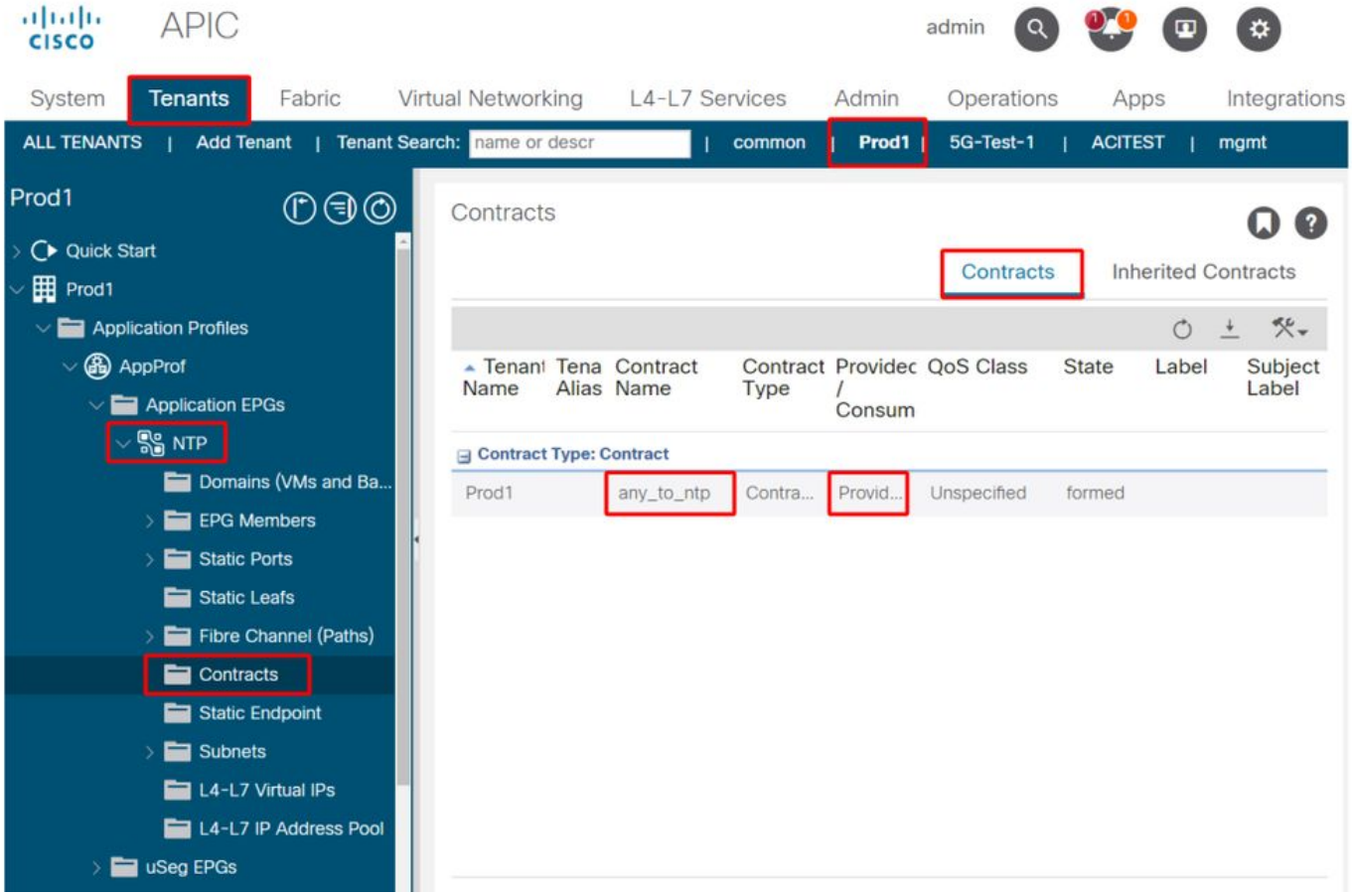
導航到VRF，並檢查「EPG Collection for VRF」下是否有已使用的合約配置為vzAny。

配置為已使用的vzVrf上的任何合約



3.驗證是否將同一合約應用為EPG NTP上提供的合約

為了建立合約關係，需要將同一合約應用為EPG NTP提供的合約，EPG NTP正在向其VRF中的其他EPG提供NTP服務。



4.使用contract_parser.py或「show zoning-rule」對入口枝葉進行分割槽規則驗證

入口枝葉應具有2個分割槽規則，以允許任何EPG和EPG NTP之間的雙向流量流（如果合約主題設定為允許兩個方向）。「任何EPG」在分割槽規則程式設計中表示為pcTag 0。

在指定VRF時，在入口枝葉上使用contract_parser.py或「show zoning-rule」命令可確保分割槽規則已程式設計。

允許流量從VRF中的其他EPG傳入/傳出EPG NTP的分割槽規則

使用contract_parser.py和「show zoning-rule」檢查是否存在基於vzAny的分割槽規則。

這裡顯然有兩種規則：

1. 規則4156和規則4168，允許Any對NTP，反之亦然。它們的優先順序為13和14: 允許流量從任何EPG(pcTag 0)流到EPG NTP(pcTag 49161)的分割槽規則。允許流量從EPG NTP(pcTag 46161)流向任何其他EPG(pcTag 0)的分割槽規則。
2. 規則4165，它是優先順序為21的any to any deny規則（預設）。

鑑於最低優先順序具有優先順序，VRF的所有EPG都將具有訪問NTP EPG的許可權。

```
fab3-leaf8# contract_parser.py --vrf Prod1:VRF
Key:
[prio:RuleId] [vrf:{str}] action protocol src-epg [src-l4] dst-epg [dst-l4]
[flags][contract:{str}] [hit=count]
```

```
[13:4156] [vrf:Prod1:VRF1] permit ip tcp tn-Prod1/ap-Services/epg-NTP(49161) eq 123 epg:any
[contract:uni/tn-Prod1/brc-any_to_ntp] [hit=0]
[14:4168] [vrf:Prod1:VRF1] permit ip tcp epg:any tn-Prod1/ap-Services/epg-NTP(49161) eq 123
[contract:uni/tn-Prod1/brc-any_to_ntp] [hit=0]
[16:4176] [vrf:Prod1:VRF1] permit any epg:any tn-Prod1/bd-App(16386) [contract:implicit] [hit=0]
[16:4174] [vrf:Prod1:VRF1] permit any epg:any tn-Prod1/bd-Services(32776) [contract:implicit]
[hit=0]
[16:4160] [vrf:Prod1:VRF1] permit arp epg:any epg:any [contract:implicit] [hit=0]
[21:4165] [vrf:Prod1:VRF1] deny,log any epg:any epg:any [contract:implicit] [hit=65]
[22:4164] [vrf:Prod1:VRF1] deny,log any epg:any pfx-0.0.0.0/0(15) [contract:implicit] [hit=0]

fab3-leaf8# show zoning-rule scope 2654209
+---------+--------+--------+----------+---------+---------+---------+------------+----------+--
--------------------+
| Rule ID | SrcEPG | DstEPG | FilterID |   Dir   |  operSt |  Scope  |    Name    |  Action
|       Priority       |
+---------+--------+--------+----------+---------+---------+---------+------------+----------+--
--------------------+
|   4165  |   0    |   0    | implicit | uni-dir | enabled | 2654209 |            | deny,log |
any_any_any(21)    |
|   4160  |   0    |   0    | implarp  | uni-dir | enabled | 2654209 |            |  permit  |
any_any_filter(17)  |
|   4164  |   0    |   15   | implicit | uni-dir | enabled | 2654209 |            | deny,log |
any_vrf_any_deny(22) |
|   4176  |   0    | 16386  | implicit | uni-dir | enabled | 2654209 |            |  permit  |
any_dest_any(16)   |
|   4174  |   0    | 32776  | implicit | uni-dir | enabled | 2654209 |            |  permit  |
any_dest_any(16)   |
|   4168  |   0    | 49161  |   424    | uni-dir | enabled | 2654209 | any_to_ntp |  permit  |
any_dest_filter(14)  |
|   4156  | 49161  |   0    |   425    | uni-dir | enabled | 2654209 | any_to_ntp |  permit  |
src_any_filter(13)  |
+---------+--------+--------+----------+---------+---------+---------+------------+----------+--
--------------------+
```

共用L3Out到EPG

關於共用L3Out

共用第3層輸出是一種配置，它允許在一個VRF中有一個L3Out提供一些服務（外部訪問），並且一個或多個其他VRF使用此L3Out。有關共用L3Out的詳細資訊，請參閱「外部路由」一章。

執行共用L3Out時，建議讓合約的提供者為共用L3Out，將EPG為合約的消費者。本節將說明此情境。

建議不要採取相反的做法，即L3Out使用EPG提供的服務。原因與可擴充性有關，因為對於共用服務，分割槽規則只安裝在使用者VRF上。消費和提供原則表示流量從何處開始。使用預設入口策略實施時，這意味著策略實施將應用於消費者端，更具體地說是應用於入口枝葉（非邊界枝葉）。 入口枝葉要強制實施策略，需要目標的pcTag。在此案例中，目標是外部EPG pcTag。入口枝葉因此執行策略實施並將資料包轉發到邊界枝葉。邊界枝葉在其交換矩陣鏈路上接收資料包，該鏈路執行路由查詢(LPM)並將資料包轉發到目的字首的鄰接關係。

但是，在將流量傳送到目標EP時，邊界枝葉不會執行任何策略實施，在返回流量返回到源EP時也不會執行任何策略實施。

因此，只有入口非BL枝葉的策略CAM安裝了條目（在消費者VRF中），BL的策略CAM不會受到影響。

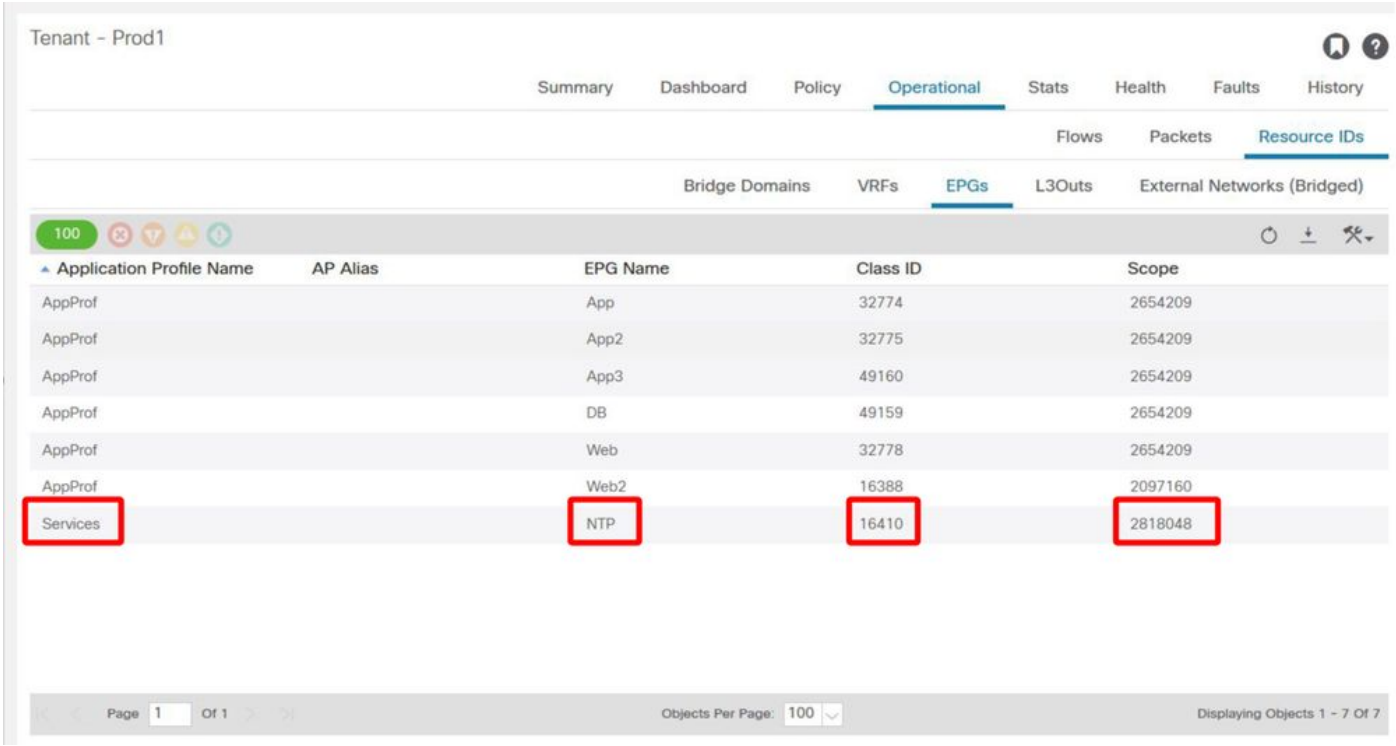
排除共用L3out故障

工作流程

1.驗證消費者EPG的EPG pcTag和VRF VNID/範圍

使用共用L3Out時，分割槽規則僅安裝在使用者VRF中。提供商必須具有允許在所有消費者VRF中使用此pcTag的全域性pcTag（低於16k）。在我們的場景中，提供商是外部EPG，將具有全域性pcTag。與往常一樣，消費者EPG將具有本地pcTag。

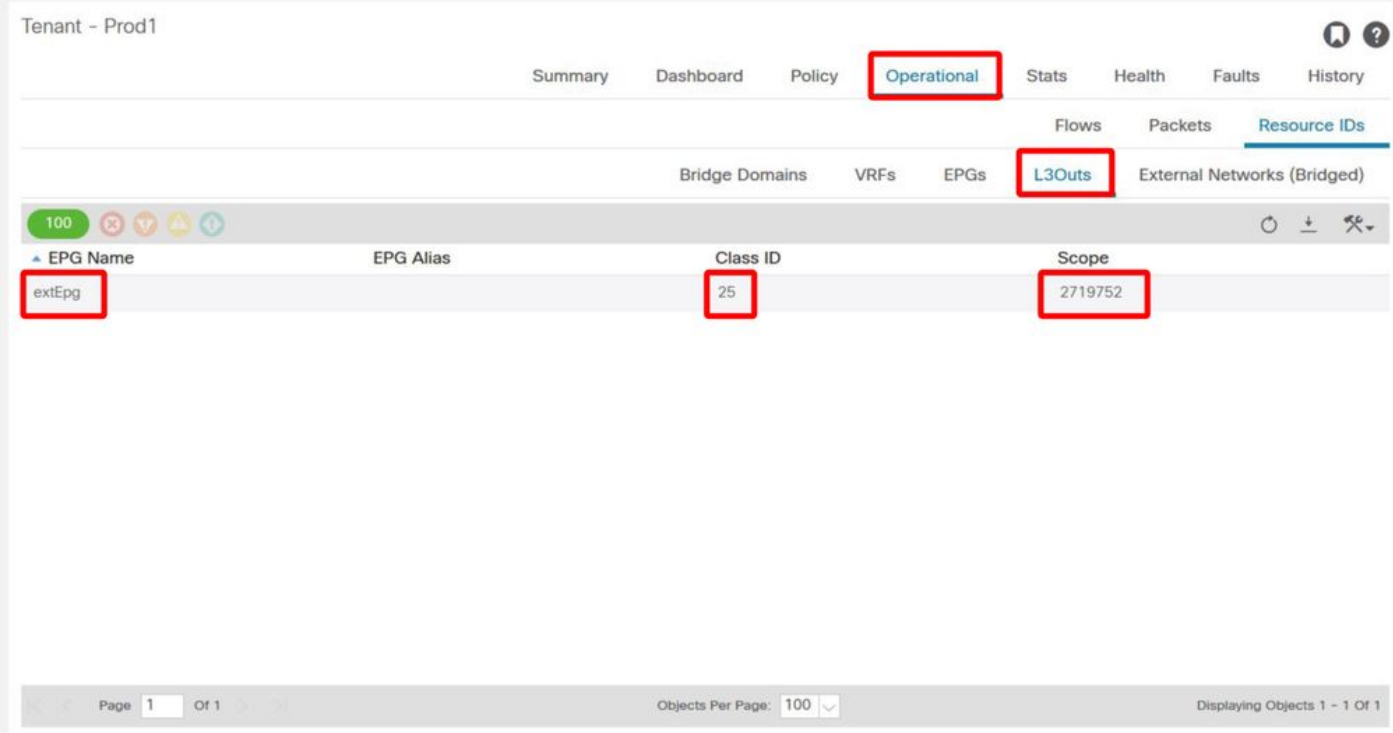
消費者EPG的pcTag



2.驗證提供商L3Out EPG的pcTag和VRF VNID/範圍

如步驟1所述，提供商L3Out EPG具有全域性範圍pcTag作為L3Out的字首，它們洩漏到消費者VRF中。因此，L3Out EPG pcTag必須不與使用者VRF中的pcTag重疊，因此它位於全域性pcTag範圍內。

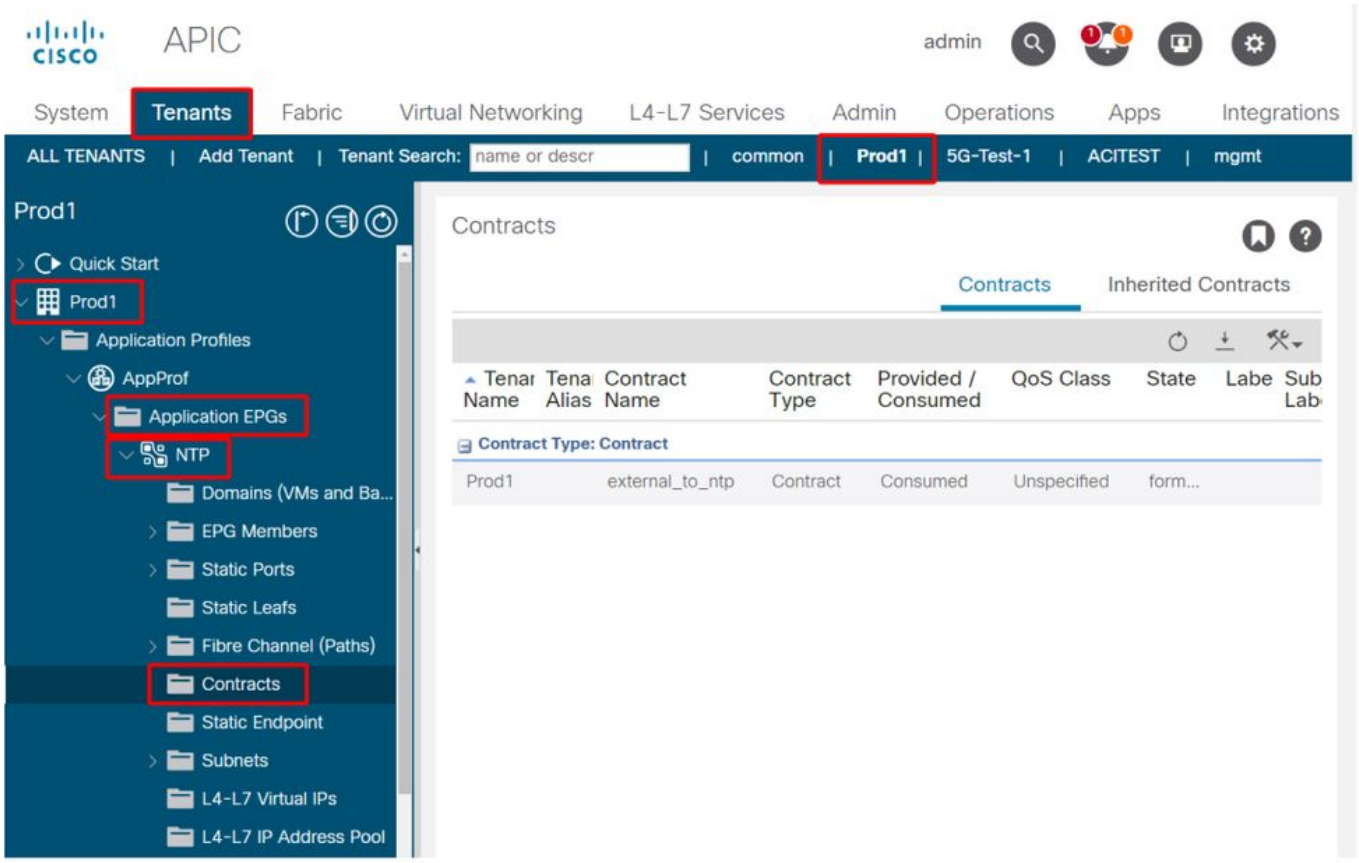
提供商外部EPG的pcTag



3.驗證使用者EPG是否已配置匯入租戶範圍的合約或全域性合約

在EPG/BD下定義子網的消費者EPG NTP正在使用「租戶」或「全域性」範圍合約

EPG使用的合約



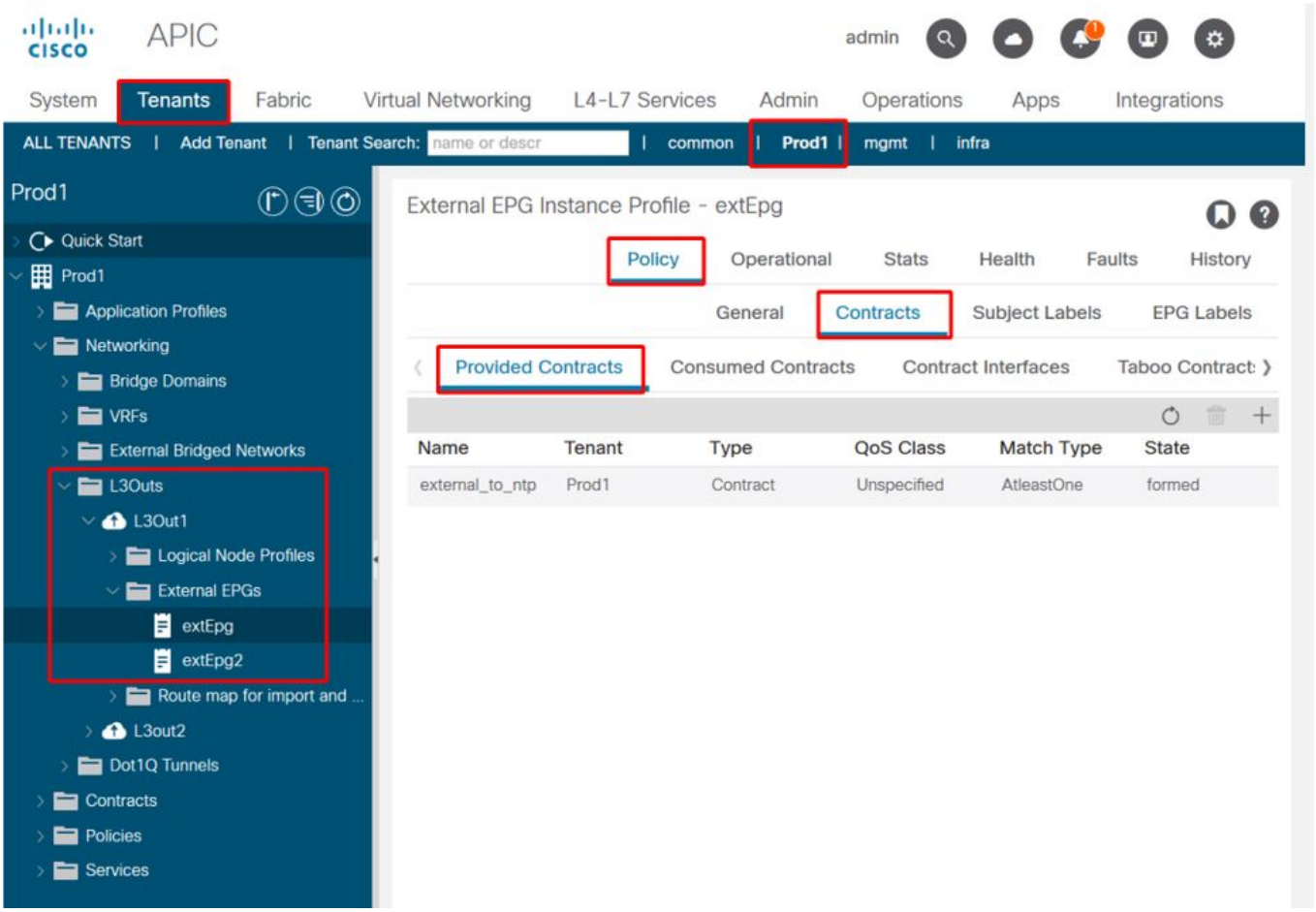
4.驗證使用者EPG的BD是否配置了作用域設定為「VRF之間共用」的子網

EPG的子網在網橋域下配置，但必須具有「在VRF之間共用」標誌（允許路由洩漏）和「通告外部」標誌（允許通告到L3Out）

5.驗證提供商L3Out EPG是否已配置匯入租戶範圍合約或全域性合約

L3Out EPG應具有租戶範圍的合約或配置為提供的合約的全域性合約。

提供程式L3Out上的合約



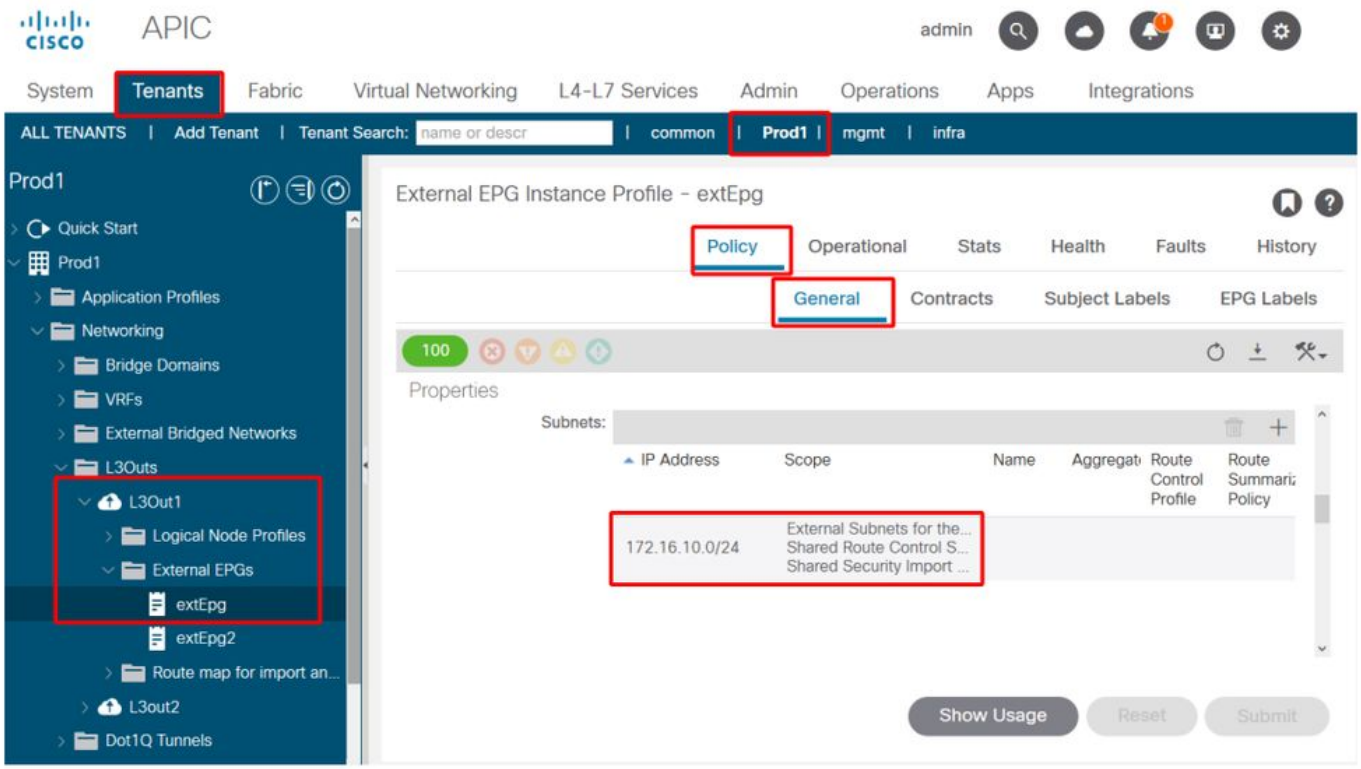
6.驗證提供程式L3Out EPG是否配置了已檢查必要範圍的子網

提供程式L3Out EPG應該具有配置以下範圍的「待洩漏」字首：

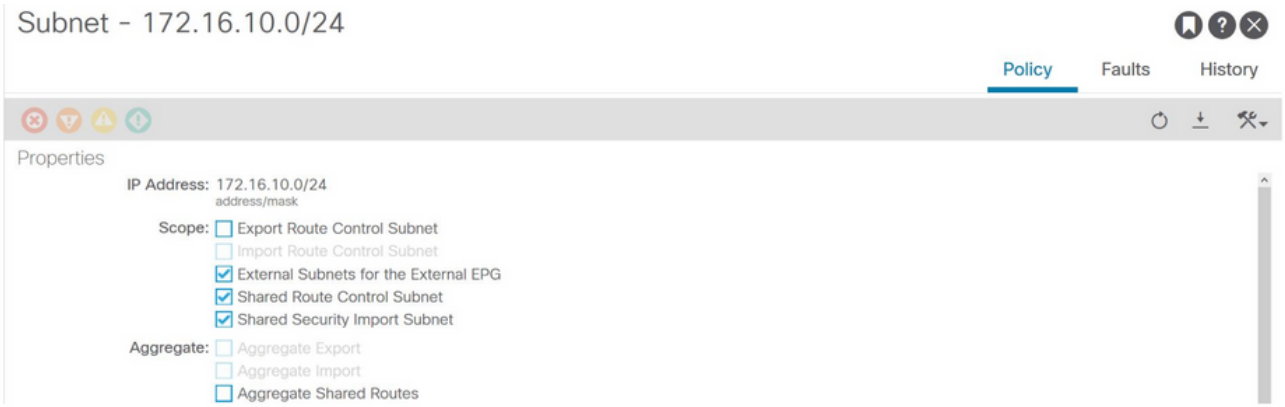
- 外部EPG的外部子網。
- 共用路由控制子網。
- 共用安全匯入子網。

有關L3Out EPG中子網標誌的詳細資訊，請參閱「外部轉發」一章。

外部EPG子網設定



已擴展外部EPG子網設定



7.驗證使用者VRF的非BL上L3Out EPG子網的pcTag

當目的地為外部EPG子網的流量進入非BL時，會根據目的地字首執行查詢以確定pcTag。可以在非BL上使用以下命令檢查這一點。

請注意，此輸出在VNI範圍(用2818048VRF VNID)內執行。通過檢視該表，消費者可以找到目標的pcTag，即使它不在同一個VRF中。

```
fab3-leaf8# vsh -c 'show system internal policy-mgr prefix' | egrep 'Vrf-Vni|==|common:default'
Vrf-Vni VRF-Id Table-Id Table-State  VRF-Name
Addr                                Class Shared Remote Complete
======= ======  ========== =======  ============================
================================ ====== ====== ====== ========
2818048 19     0x13          Up     common:default
0.0.0.0/0   15      False  False  False
2818048 19     0x80000013    Up     common:default
::/0   15      False  False  False
2818048 19     0x13          Up     common:default
172.16.10.0/24  25      True   True   False
```

上面的輸出顯示了L3Out EPG子網及其全域性pcTag 25的組合。

8.驗證消費者VRF的非BL上的已程式設計分割槽規則

使用「contract_parser.py」或「show zoning-rule」命令並指定VRF。

下面的命令輸出顯示了兩個分割槽規則，這些規則允許從使用者EPG本地pcTag 16410到L3Out EPG全域性pcTag 25的流量。這屬於範圍2818048，即使用者VRF的範圍。

```
fab3-leaf8# show zoning-rule scope 2818048
+---------+--------+--------+----------+----------------+---------+---------+----------------+-
---------+----------------------+
| Rule ID | SrcEPG | DstEPG | FilterID |      Dir       |  operSt |  Scope  |       Name     |
Action  |        Priority      |
+---------+--------+--------+----------+----------------+---------+---------+----------------+-
---------+----------------------+
|   4174  |   0    |   0    | implarp  |    uni-dir     | enabled | 2818048 |                |
permit  |   any_any_filter(17)  |
|   4168  |   0    |   15   | implicit |    uni-dir     | enabled | 2818048 |                |
deny,log |  any_vrf_any_deny(22) |
|   4167  |   0    | 32789  | implicit |    uni-dir     | enabled | 2818048 |                |
permit  |    any_dest_any(16)   |
|   4159  |   0    |   0    | implicit |    uni-dir     | enabled | 2818048 |                |
deny,log |    any_any_any(21)    |
|   4169  |   25   |   0    | implicit |    uni-dir     | enabled | 2818048 |                |
deny,log | shsrc_any_any_deny(12)|
|   4156  |   25   | 16410  |   425    | uni-dir-ignore | enabled | 2818048 | external_to_ntp |
permit  |     fully_qual(7)     |
|   4131  | 16410  |   25   |   424    |     bi-dir     | enabled | 2818048 | external_to_ntp |
permit  |     fully_qual(7)     |
+---------+--------+--------+----------+----------------+---------+---------+----------------+-
---------+----------------------+

fab3-leaf8# contract_parser.py  --vrf common:default
Key:
[prio:RuleId] [vrf:{str}] action protocol src-epg [src-l4] dst-epg [dst-l4]
[flags][contract:{str}] [hit=count]

[7:4131] [vrf:common:default] permit ip tcp tn-Prod1/ap-Services/epg-NTP(16410) tn-Prod1/l3out-
L3Out1/instP-extEpg(25) eq 123  [contract:uni/tn-Prod1/brc-external_to_ntp] [hit=0]
[7:4156] [vrf:common:default] permit ip tcp tn-Prod1/l3out-L3Out1/instP-extEpg(25) eq 123 tn-
Prod1/ap-Services/epg-NTP(16410)  [contract:uni/tn-Prod1/brc-external_to_ntp] [hit=0]
[12:4169] [vrf:common:default] deny,log any tn-Prod1/l3out-L3Out1/instP-extEpg(25) epg:any
[contract:implicit] [hit=0]
[16:4167] [vrf:common:default] permit any epg:any tn-Prod1/bd-Services(32789)
[contract:implicit] [hit=0]
[16:4174] [vrf:common:default] permit arp epg:any epg:any [contract:implicit] [hit=0]
[21:4159] [vrf:common:default] deny,log any epg:any epg:any [contract:implicit] [hit=0]
[22:4168] [vrf:common:default] deny,log any epg:any pfx-0.0.0.0/0(15) [contract:implicit]
[hit=0]
```

9.驗證提供商VRF的BL上的已程式設計分割槽規則

使用「contract_parser.py」或「show zoning-rule」命令並指定VRF。以下命令輸出顯示提供程式VRF中沒有**NO specific zoning-rules**（如之前多次描述的那樣）。

它位於提供商2719752的範圍之內。

```
border-leaf# show zoning-rule scope 2719752
```

```
+---------+--------+--------+----------+----------------+---------+---------+-------------+----
------+----------------------+
| Rule ID | SrcEPG | DstEPG | FilterID |      Dir       |  operSt |  Scope  |     Name    |
Action  |       Priority       |
+---------+--------+--------+----------+----------------+---------+---------+-------------+----
------+----------------------+
|   4134  | 10937  |   24   | default  | uni-dir-ignore | enabled | 2719752 | vrf1_to_vrf2 |
permit  |    src_dst_any(9)    |
|   4135  |   24   | 10937  | default  |     bi-dir     | enabled | 2719752 | vrf1_to_vrf2 |
permit  |    src_dst_any(9)    |
|   4131  |   0    |   0    | implicit |    uni-dir     | enabled | 2719752 |              |
deny,log |   any_any_any(21)    |
|   4130  |   0    |   0    | implarp  |    uni-dir     | enabled | 2719752 |              |
permit  |  any_any_filter(17)  |
|   4132  |   0    |   15   | implicit |    uni-dir     | enabled | 2719752 |              |
deny,log | any_vrf_any_deny(22) |
+---------+--------+--------+----------+----------------+---------+---------+-------------+----
------+----------------------+

border-leaf# contract_parser.py --vrf Prod1:VRF3
Key:
[prio:RuleId] [vrf:{str}] action protocol src-epg [src-l4] dst-epg [dst-l4]
[flags][contract:{str}] [hit=count]

[9:4134] [vrf:Prod1:VRF3] permit any tn-Prod1/l3out-L3Out1/instP-extEpg2(10937) tn-Prod1/l3out-
L3Out2/instP-extEpg2(24) [contract:uni/tn-Prod1/brc-vrf1_to_vrf2] [hit=0]
[9:4135] [vrf:Prod1:VRF3] permit any tn-Prod1/l3out-L3Out2/instP-extEpg2(24) tn-Prod1/l3out-
L3Out1/instP-extEpg2(10937) [contract:uni/tn-Prod1/brc-vrf1_to_vrf2] [hit=0]
[16:4130] [vrf:Prod1:VRF3] permit arp epg:any epg:any [contract:implicit] [hit=0]
[21:4131] [vrf:Prod1:VRF3] deny,log any epg:any epg:any [contract:implicit] [hit=0]
[22:4132] [vrf:Prod1:VRF3] deny,log any epg:any pfx-0.0.0.0/0(15) [contract:implicit] [hit=0]
```

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）。