ACI 보안 정책 문제 해결 - 계약

목차

소개

이 문서에서는 ACI 보안 정책(계약이라고 함)을 이해하고 문제를 해결하는 단계를 설명합니다.

배경 정보

이 문서의 자료는 Troubleshooting Cisco Application Centric Infrastructure, Second Edition 책, 특히 Security Policies(보안 정책) - Overview(개요), Security Policies(보안 정책) - Tools(툴), Security Policies(보안 정책) - EPG to EPG(EPG to EPG), Security Policies(**보안 정책) - Preferred group(선호 그룹)** 및 Security Policies(보안 정책) - vzAny to EPG(EPG) 장에서 추출했습니다**.**

개요

ACI 솔루션의 기본 보안 아키텍처는 허용 목록 모델을 따릅니다. VRF가 **비시행** 모드로 구성되지 않는 한 **EPG** 트래픽 흐름에 대한 모든 EPG는 암시적으로 삭제됩니다. 기본 허용 목록 모델에서 암시하는 것처럼 기본 VRF 설정은 **강제** 모드입니다. 스위치 노드에서 zoning-rule을 구현하여 트래픽 흐름을 허용하거나 명시적으로 거부할 수 있습니다. 이러한 zoning-rule은 엔드포인트 그룹(EPG) 간의 원하는 통신 흐름과 이를 정의하는 데 사용되는 방법에 따라 다양한 구성으로 프로그래밍할 수 있습니다. zoning-rule 엔트리는 스테이트풀(stateful)이 아니며, 일반적으로 규칙이 프로그래밍되면 2개의 EPG가 제공된 포트/소켓에 따라 허용/거부됩니다.

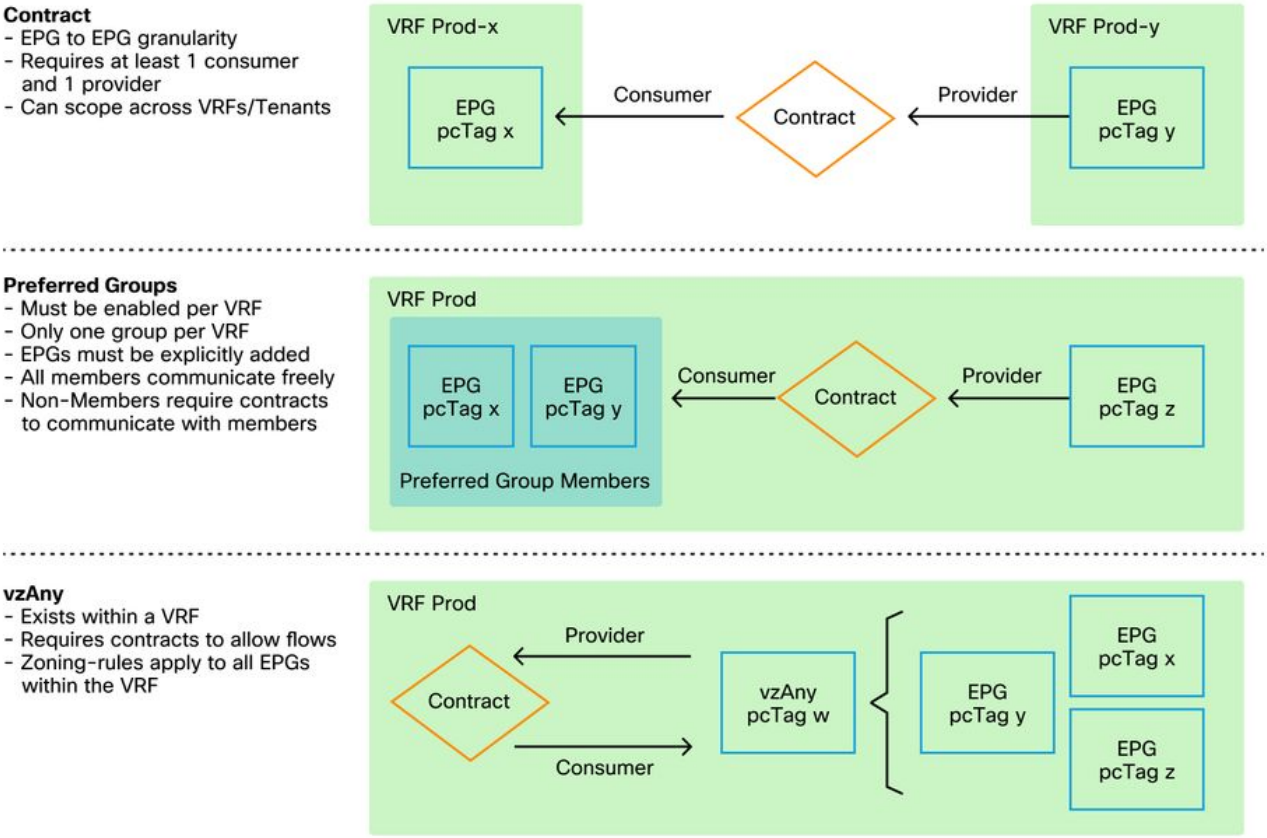
zoning-rule을 프로그래밍하는 방법

ACI 내에서 zoning-rule을 프로그래밍하는 주요 방법은 다음과 같습니다.

- **EPG-EPG 계약: 일반적으로** 둘 이상의 개별 엔드포인트 그룹에 영역 지정 규칙을 프로그래밍하려면 하나 이상의 소비자와 공급자가 필요합니다.
- **선호 그룹:** VRF 레벨에서 그룹화를 활성화해야 합니다. VRF당 하나의 그룹만 존재할 수 있습니다. 그 그룹의 모든 구성원들은 자유롭게 의사소통을 할 수 있다. 비구성원은 계약을 통해 선호 그룹에 대한 플로우를 허용해야 합니다.
- **vzAny:** 지정된 VRF 아래에 정의된 'EPG 컬렉션'입니다. vzAny는 VRF의 모든 EPG를 나타냅니다. vzAny를 사용하면 하나의 계약 연결을 통해 VRF 내의 모든 EPG와 하나의 EPG 간에 흐름이 허용됩니다.

다음 다이어그램은 위의 각 방법이 제어할 수 있는 zoning-rule의 세분화를 참조하는 데 사용할 수 있습니다.

영역 지정 규칙 방법론 간 비교



zoning-rule을 프로그래밍하는 계약방식을 활용하면서 계약범위를 정의할 수 있는 옵션이 있다. 이 옵션은 경로 유출/공유 서비스 설계가 필요한 경우 신중하게 고려해야 합니다. ACI 패브릭 내에서 한 VRF에서 다른 VRF로 전환하려는 경우 계약이 이를 위한 방법입니다.

범위 값은 다음과 같습니다.

- **애플리케이션:** 계약 소비자/공급자 관계는 동일한 애플리케이션 프로파일 내에 정의된 EPG 간의 프로그램 규칙만 수행합니다. 다른 애플리케이션 프로파일 EPG에서 동일한 계약을 재사용하면 둘 사이의 누화가 허용되지 않습니다.
- **VRF(기본값):** 계약 소비자/공급자 관계는 동일한 VRF 내에 정의된 EPG 간의 규칙을 프로그래밍합니다. 다른 애플리케이션 프로파일 EPG에서 동일한 계약을 다시 사용하면 둘 사이의 누화가 가능합니다. 원하는 플로우만 허용되도록 주의하십시오. 그렇지 않으면 의도하지 않은 누화를 방지하기 위해 새 계약을 정의해야 합니다.

- **테넌트:** 계약 소비자/공급자 관계는 동일한 테넌트 내에 정의된 EPG 간의 규칙을 프로그래밍합니다. 단일 테넌트 내에서 여러 VRF에 연결된 EPG가 있고 동일한 계약을 소비/제공하는 경우 이 범위를 사용하여 VRF 간 통신을 허용하기 위해 경로 유출을 유도할 수 있습니다.
- **글로벌:** 계약 소비자/공급자 관계는 ACI 패브릭 내의 모든 테넌트에서 EPG 간의 규칙을 프로그래밍합니다. 이는 가장 높은 범위의 정의이며, 의도하지 않은 흐름 누출을 방지하기 위해 이전에 정의된 계약에서 이를 활성화할 경우 주의해야 합니다.

영역 지정 규칙 항목 읽기

zoning-rule이 프로그래밍되면 leaf에서 다음과 같이 나타납니다.

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

- **규칙 ID:** 규칙 항목의 ID입니다. 고유한 식별자로 작동하는 것 외에 실제적인 의미는 없습니다.
- **소스 EPG:** 소스 엔드포인트 그룹의 VRF당 고유한 ID(pcTag)입니다.
- **Dst EPG:** 대상 엔드포인트 그룹의 VRF(pcTag)당 고유한 ID입니다.
- **FilterID:** 규칙이 매칭을 시도하는 필터의 ID입니다. Filter에는 규칙이 매칭할 프로토콜 정보가 포함됩니다.
- **Dir:** zoning-rule의 방향성.
- **OperSt:** 규칙의 작동 상태입니다.
- **범위:** 규칙이 매칭할 VRF의 고유 ID입니다.
- **이름:** 해당 항목이 프로그래밍된 계약의 이름입니다.
- **Action:** 해당 항목과 일치할 때 Leaf가 수행하는 작업 포함 내용: [삭제, 허용, 로그, 리디렉션].
- **우선순위:** 일치하는 범위, SrcEPG, DstEPG 및 필터 항목이 있는 경우 작업에 대해 영역 지정 규칙이 검증되는 순서.

정책 CAM(Content-Addressable Memory)

각 조닝 규칙이 프로그래밍되면 필터 항목에 대해 매핑된 조닝 규칙 항목의 행렬이 스위치에서 **Policy CAM**을 사용하기 시작합니다. ACI 패브릭을 통해 허용되는 플로우를 설계하는 동안, 최종 설계에 따라 계약을 새로 생성하는 것이 아니라 계약을 재사용할 때 특별히 주의해야 합니다. 결과적으로 조닝 규칙을 이해하지 못한 채 여러 EPG에서 동일한 계약을 재사용하면 예기치 않게 여러 플로우에 빠르게 캐스케이딩될 수 있습니다. 동시에 이러한 의도하지 않은 흐름은 계속해서 정책 CAM을 사용합니다. 정책 CAM이 가득 차면 조닝 규칙 프로그래밍이 실패하기 시작하며, 이는 컨피그레이션 및 엔드포인트 동작에 따라 예상치 못한 일시적인 손실을 초래할 수 있습니다.

공유 L3Out의 VRF 유출, 글로벌 pcTag 및 정책 시행 방향

이는 계약을 구성해야 하는 공유 서비스 활용 사례에 대한 특별 설명입니다. 공유 서비스는 일반적으로 '테넌트' 또는 '글로벌' 범위 계약의 사용에 의존하는 ACI 패브릭 내의 VRF 간 트래픽을 의미합니다. 이를 완전히 이해하려면 먼저 EPG에 할당되는 일반적인 pcTag 값이 전체적으로 고유하지 않다는 생각을 강화해야 합니다. pcTag는 VRF로 범위가 지정되며 동일한 pcTag를 다른 VRF 내에서 재사용할 수 있습니다. 경로 유출에 대한 논의가 시작되면 서브넷 및 pcTag를 비롯한 전역적으로 고유한 값이 필요하다는 점을 포함하여 ACI 패브릭에 대한 요구 사항을 시행하기 시작합니다.

이를 특별한 고려 사항으로 만드는 것은 EPG가 소비자 대 공급자일 때 관련된 방향성 측면입니다. 공유 서비스 시나리오에서 공급자는 일반적으로 패브릭 고유 값을 얻기 위해 글로벌 pcTag를 구동해야 합니다. 동시에 소비자는 VRF 범위가 지정된 pcTag를 보유하여 이제 프로그래밍하고 정책을 시행하기 위해 전역 pcTag 값의 사용을 이해할 수 있도록 특수한 위치에 배치합니다.

참고로, pcTag 할당 범위는 다음과 같습니다.

- 시스템 예약: 1-15 .
- 전역 범위: 공유 서비스 공급자 EPG의 경우 16-16384.
- 로컬 범위: VRF 범위 EPG의 16385-65535.

VRF 정책 제어 시행 방향

각 VRF에서 시행 방향 설정을 정의할 수 있습니다.

- 시행 방향의 기본 설정은 Ingress입니다.
- 시행 방향의 다른 옵션은 이그레스(Egress)입니다.

정책이 시행되는 위치를 파악하는 것은 여러 가지 변수에 따라 달라집니다.

아래 표에는 리프 레벨에서 보안 정책이 적용되는 위치가 나와 있습니다.

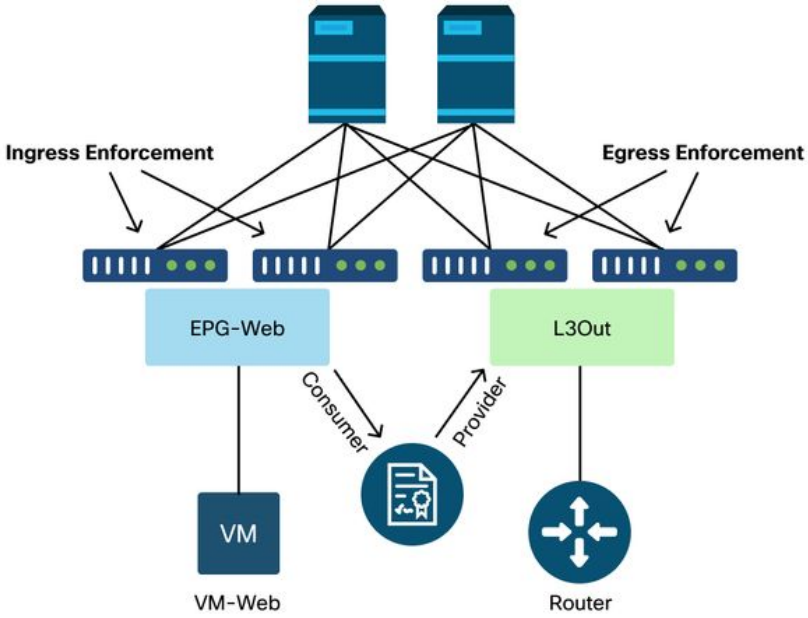
정책이 적용되는 위치

시나리오	VRF 시행 모드	소비자	공급자	정책 적용 대상
VRF 내	인그레스/이그레스	이피지	이피지	·대상 엔드포인트를 학습한 경우: 인그레스 리프* ·대상 엔드포인트를 학습하지 않은 경우: 이그레스 리프
	인그레스	이피지	L3Out EPG	소비자 리프(비경계 리프)
	인그레스	L3Out EPG	이피지	공급자 리프(비경계 리프)
	이그레스	이피지	L3Out EPG	경계 리프 -> 비경계 리프 트래픽 ·대상 엔드포인트를 학습한 경우: 보더 리프
	이그레스	L3Out EPG	이피지	·대상 엔드포인트를 학습하지 않은 경우: 비경계 리프 비경계 리프-> 경계 리프 트래픽 ·보더 리프
	인그레스/이그레스	L3Out EPG	L3Out EPG	인그레스 리프*
VRF 간	인그레스/이그레스	이피지	이피지	소비자 리프
	인그레스/이그레스	이피지	L3Out EPG	소비자 리프(비경계 리프)
	인그레스/이그레스	L3Out EPG	이피지	인그레스 리프*
	인그레스/이그레스	L3Out EPG	L3Out EPG	인그레스 리프*

*정책 시행은 패킷이 도달한 첫 번째 leaf에 적용됩니다.

아래 그림에는 EPG-Web을 소비자로, L3Out EPG를 제공자로 사용하는 contract enforcement의 예가 나와 있습니다. VRF가 인그레스 시행 모드로 설정된 경우 EPG-Web이 상주하는 리프 노드에 의해 정책이 시행됩니다. VRF가 이그레스(Egress) 시행 모드로 설정된 경우, VM-Web 엔드포인트가 보더 리프에 학습된 경우 L3Out이 상주하는 보더 리프 노드에 의해 정책이 시행됩니다.

인그레스 시행 및 이그레스 시행



툴

정책 삭제를 식별하는 데 도움이 되는 다양한 툴과 명령이 **있습니다**. 정책 삭제는 계약 컨피그레이션 또는 부족으로 인한 패킷 삭제로 정의할 수 있습니다.

영역 지정 규칙 검증

완료된 계약 소비자/공급자 관계의 결과로 리프 스위치에 프로그래밍된 zoning-rule을 명시적으로 검증하는 데 다음 툴과 명령을 사용할 수 있습니다.

'영역 지정 규칙 표시'

모든 영역 지정 규칙을 표시하는 스위치 레벨 명령.

```
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)        |
```

```
+---------+--------+--------+----------+---------------+---------+----------+----------------
+----------+---------------------------+
```

'영역 지정 필터 표시'

영역 지정 규칙이 작동하는 스포츠/포트 정보를 포함하는 필터. 이 명령으로 필터 프로그래밍을 확인할 수 있습니다.

```
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   |
+----------+----------+-------------+-------------+-------------+----------+------------+------
-------+-------------+-------------+----------+
```

'시스템 내부 정책 관리자 통계 표시'

zoning-rule당 적중 횟수를 확인하기 위해 이 명령을 실행할 수 있습니다. 이는 우선 순위가 더 높을 수 있는 암시적 삭제 규칙과 같이 예상되는 규칙이 다른 규칙에 비해 적중되는지를 확인하는 데 유용합니다.

```
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'

ACL(계약) 관련 삭제 및 플로우 관련 정보를 보고하는 Bash 레벨에서 실행할 수 있는 스위치 레벨 명령으로서, 다음과 같습니다.

- VRF
- VLAN-ID
- 소스 MAC/대상 MAC
- 소스 IP/대상 IP
- 소스 포트/대상 포트
- 소스 인터페이스

```
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

ID에서 이름 조회를 수행하는 동안 영역 지정 규칙의 상관관계를 분석하고, 필터링하고, 통계를 확인하는 출력을 생성하는 디바이스 내 Python 스크립트. 이 스크립트는 다단계 프로세스를 거쳐 특정 EPG/VRF 또는 기타 계약 관련 값으로 필터링할 수 있는 단일 명령으로 전환된다는 점에서 매우 유용합니다.

```
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]
```

패킷 분류 검증

엘람

삭제된 패킷의 경우 삭제 이유를 나타내는 전달 세부 정보를 확인하는 데 사용되는 ASIC 레벨 보고서. 이 섹션과 관련하여 사유는 SECURITY_GROUP_DENY(계약 정책 삭제)일 수 있습니다.

분류

ELAM으로 엔드 투 엔드 패킷 흐름을 추적할 수 있는 APIC의 Python 기반 유틸리티입니다.

ELAM Assistant 앱

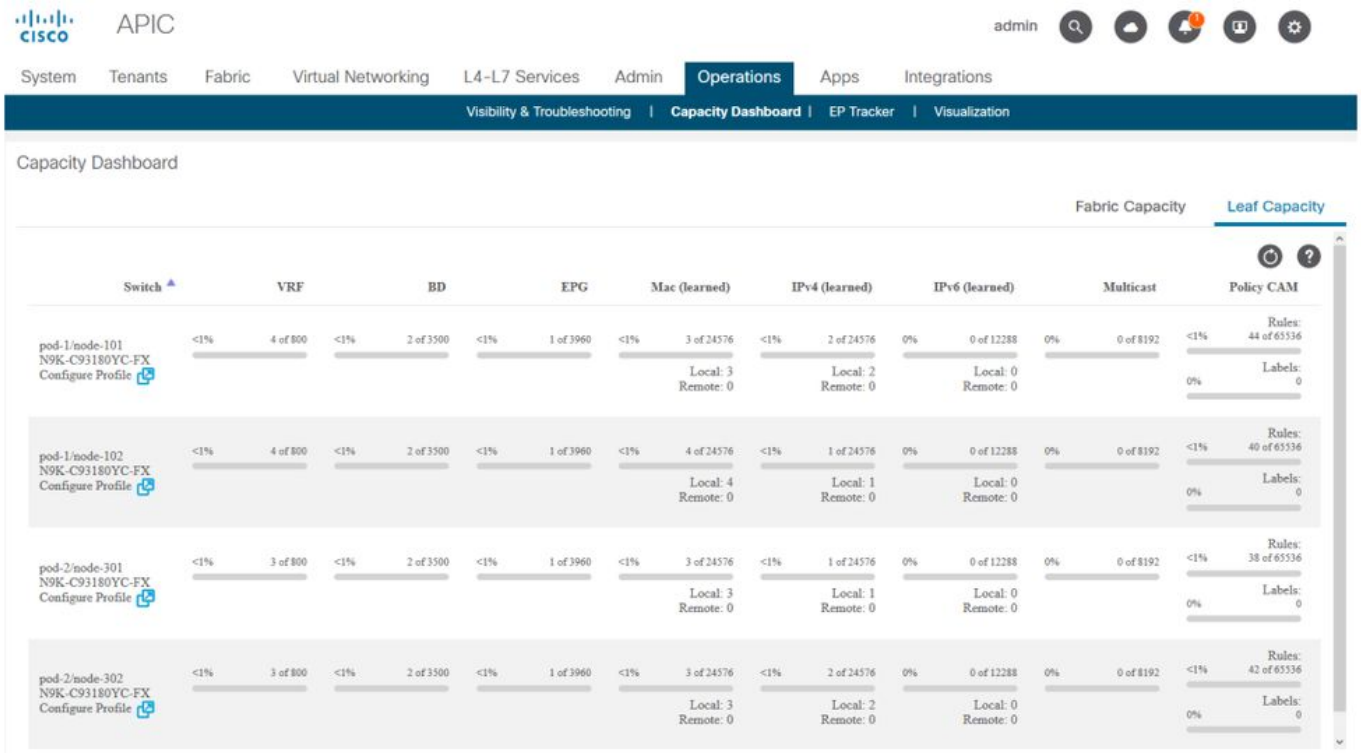
다양한 ASIC의 복잡성을 추상화하여 전달 결정 검사를 훨씬 편리하고 편리하게 해주는 APIC 앱입니다.

ELAM, Triage 및 ELAM Assistant Tools에 대한 자세한 내용은 "Intra-Fabric Forwarding" 섹션을 참조하십시오

정책 CAM 사용

리프별 정책 CAM 사용은 패브릭이 정상 상태인지 확인하기 위해 모니터링하는 중요한 매개변수입니다. 가장 빠른 모니터링 방법은 GUI에서 'Capacity Dashboard'(용량 대시보드)를 사용하고 'Policy Cam'(정책 캠) 열을 명시적으로 확인하는 것입니다.

용량 대시보드의 '리프 용량' 보기



'show platform internal hal health-stats'

이 명령은 정책 CAM을 비롯한 다양한 리소스 제한과 사용을 확인하는 데 유용합니다. 이 명령은 vsh_lc에서만 실행할 수 있으므로 iBash에서 실행하는 경우 '-c' 플래그를 사용하여 전달합니다.

```
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
=============
```

EPG에서 EPG로

일반 정책 삭제 고려 사항

두 엔드포인트 간의 연결 문제를 해결하는 방법에는 여러 가지가 있습니다. 다음 방법론은 연결 문제가 **정책 중단**(contract induced)의 결과인지 여부를 신속하고 효과적으로 격리할 수 있는 좋은 출발점을 제공합니다.

다이빙하기 전에 질문할 가치가 높은 몇 가지 질문:

- 엔드포인트가 동일한 EPG에 있습니까, 아니면 다른 EPG에 있습니까? 서로 다른 EPG(Inter-EPG)에 상주하는 두 엔드포인트 간의 트래픽은 암시적으로 거부되며 통신을 허용하기 위해 연락처가 필요합니다.동일한 EPG(intra-EPG) 내의 두 엔드포인트 간의 트래픽은 intra-EPG 격리

가 사용되지 않는 한 암시적으로 허용됩니다.
- VRF는 시행됩니까 아니면 시행되지 않습니까? VRF가 **강제** 실행 **모드**에 있는 경우, VRF 내에서 서로 다른 두 EPG의 엔드포인트가 통신을 수행하려면 계약이 필요합니다.VRF가 **비강제** 모드(VRF 내)인 경우, 적용된 ACI 계약과 상관없이 모든 트래픽이 비강제 VRF에 속하는 여러 EPG 전반에서 ACI 패브릭에 의해 허용됩니다.

방법론

사용 가능한 다양한 도구를 사용하면 영향을 받는 흐름에 대해 이미 알려진 정보 수준에 따라 다른 도구보다 더 적절하고 편리하게 시작할 수 있는 도구가 있습니다.

ACI 패브릭에서 패킷의 전체 경로를 알고 있습니까(인그레스 리프, 이그레스 리프...)?

- 대답이 예인 경우 ELAM Assistant를 사용하여 소스 또는 대상 스위치에서 삭제 이유를 식별해야 합니다.
- 아니요로 응답한 경우 Visibility & Troubleshooting, Triage, contract_parser, Operational tab in the Tenant view 및 iBash 명령을 사용하면 패킷의 경로를 좁히거나 삭제 이유를 더 자세히 파악할 수 있습니다.

Triage 툴은 이 섹션에서 자세히 설명되지 않습니다. 이 툴 사용에 대한 자세한 내용은 "Intra-Fabric Forwarding" 장을 참조하십시오.

Visibility & Troubleshooting(가시성 및 트러블슈팅)은 두 엔드포인트 간에 패킷이 삭제된 위치를 신속하게 시각화하는 데 도움이 되지만, Triage(분류)는 추가 트러블슈팅을 위해 더 자세한 정보를 표시합니다. 예: 분류 기능은 인터페이스, 삭제 이유, 영향을 받는 흐름에 대한 기타 하위 레벨 세부사항을 파악하는 데 도움이 됩니다

.

이 예제 시나리오에서는 두 엔드포인트 간의 정책 삭제 문제를 해결하는 방법을 보여 줍니다. 192.168.21.11 및 192.168.23.11

이러한 두 엔드포인트 간에 패킷 삭제를 경험한다고 가정하면 다음 트러블슈팅 워크플로를 사용하여 문제의 근본 원인을 식별합니다.

트래픽 플로우와 관련된 src/dst leaf를 식별합니다.

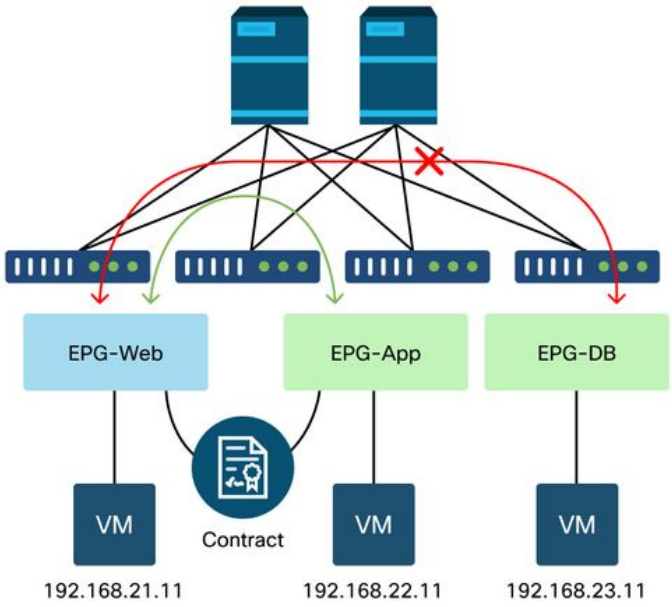
1. Visibility **& Troubleshooting(가시성 및** 문제 해결)을 사용하여 패킷 흐름을 추적하고 패킷을 삭제하는 디바이스를 식별합니다.
2. 선택한 디바이스에서 'show logging ip access-list internal packet-log deny' 명령을 실행합니다. 관심 IP 주소 중 하나가 포함된 패킷이 거부 및 로깅되는 경우 **패킷-로그는 적중별**로 관련 엔드포인트 및 계약 이름을 인쇄합니다.
3. 소스 및 대상 리프에 'contract_parser.py —vrf <tenant>:<VRF>' 명령을 사용하여 구성된 계약의 적중 횟수를 확인합니다. 패킷이 소스 또는 대상 스위치에서 계약을 적중하는 경우 관련 계약의 카운터가 증가합니다이 방법은 많은 플로우가 동일한 규칙을 통과할 수 있는 상황에서 IP access-list 내부 패킷 로그보다 덜 세분화됩니다(관심 있는 두 EPG 간의 많은 엔드포인트/플로우).

위의 단계는 다음 단락에서 더 자세히 설명합니다.

EPG에 대한 문제 해결 시나리오 예

이 예제 시나리오에서는 두 엔드포인트 간의 정책 삭제 문제를 해결하는 방법을 보여 줍니다. EPG-Web의 경우 192.168.21.11이고 EPG-DB의 경우 192.168.23.11입니다.

토폴로지

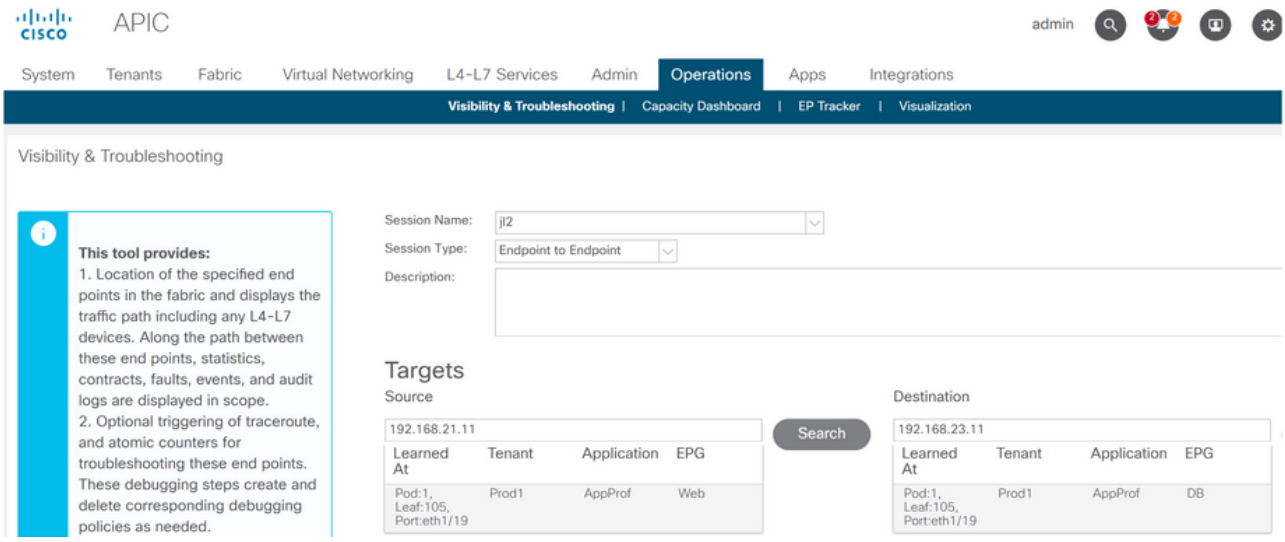


패킷 삭제와 관련된 소스 및 대상 리프 스위치 식별

가시성 및 문제 해결

Visibility & Troubleshooting(가시성 및 트러블슈팅) 도구는 특정 EP-EP 플로우에 대해 패킷 삭제가 발생한 스위치를 시각화하고 패킷이 삭제될 수 있는 위치를 식별하는 데 도움이 됩니다.

가시성 및 문제 해결 구성



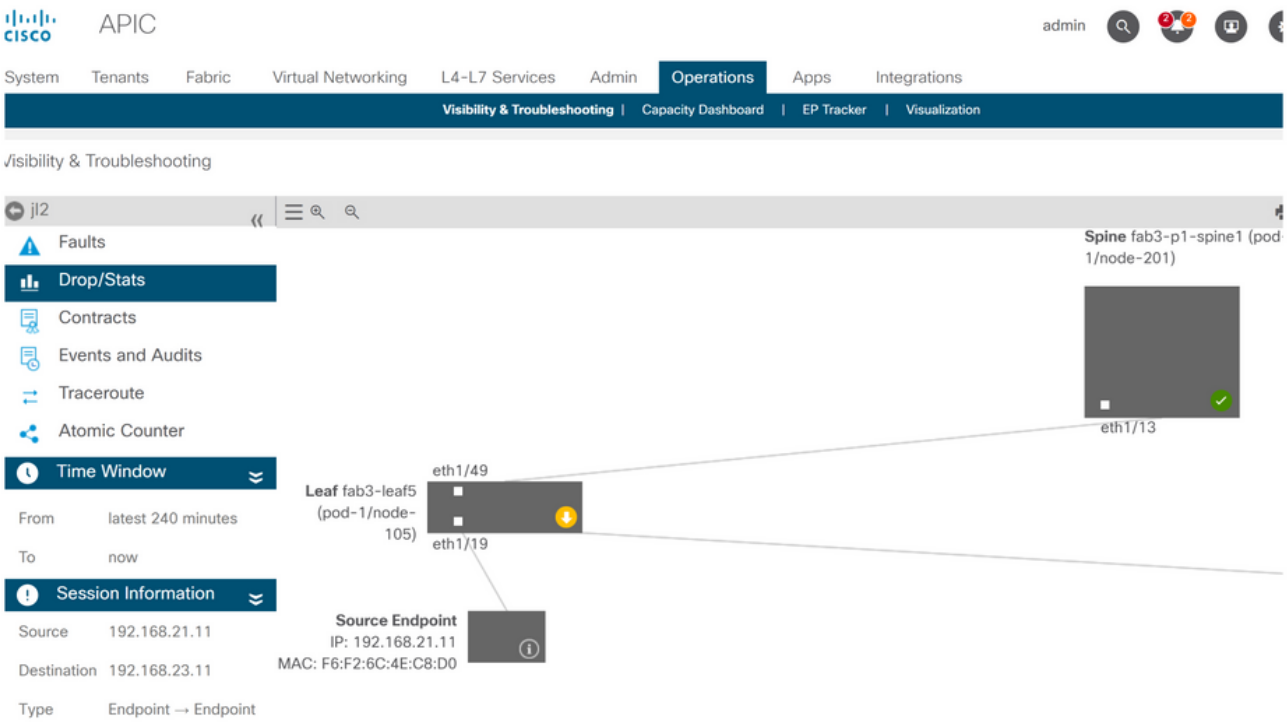
세션 이름, 소스 및 대상 엔드포인트를 구성합니다. 그런 다음 '제출' 또는 '보고서 생성'을 클릭합니다.

이 툴은 자동으로 패브릭에서 엔드포인트를 찾고 해당 EP가 속한 테넌트, 애플리케이션 프로파일

및 EPG에 대한 정보를 제공합니다.

이 경우 EP가 테넌트 Prod1에 속하고 동일한 애플리케이션 프로파일 'AppProf'에 속하며 서로 다른 EPG에 할당됨을 확인합니다. 'Web' 및 'DB'입니다.

삭제 식별

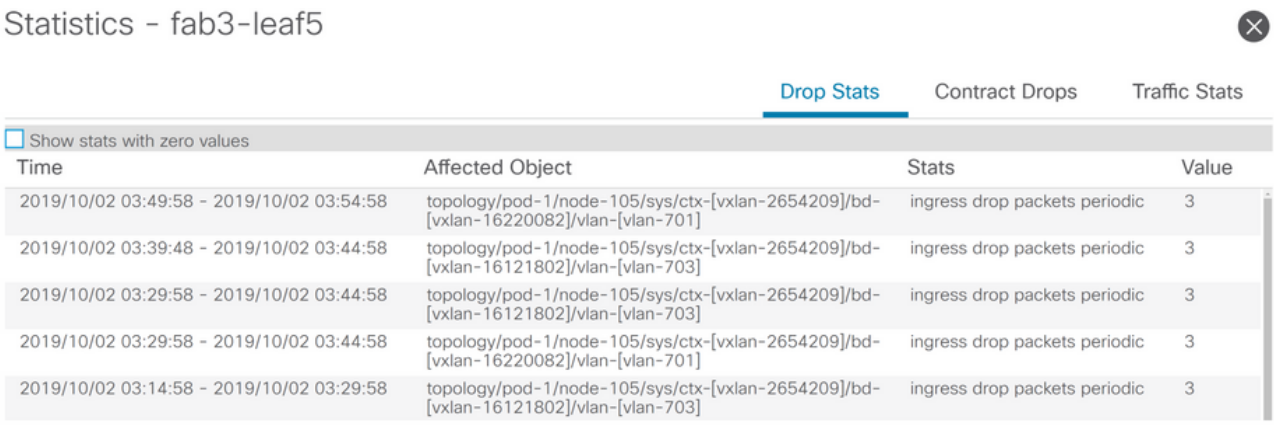


이 도구는 문제 해결 시나리오의 토폴로지를 자동으로 시각화합니다. 이 경우 두 엔드포인트가 동일한 리프 스위치에 연결됩니다.

Drop/Stats(삭제/통계) 하위 메뉴로 이동하면 해당 leaf 또는 spine에서 일반 삭제를 볼 수 있습니다. 어떤 드롭이 관련성이 있는지 파악하는 데 대한 자세한 내용은 이 책의 "패브릭 내 포워딩" 장에서 "인터페이스 드롭" 섹션을 참조하십시오.

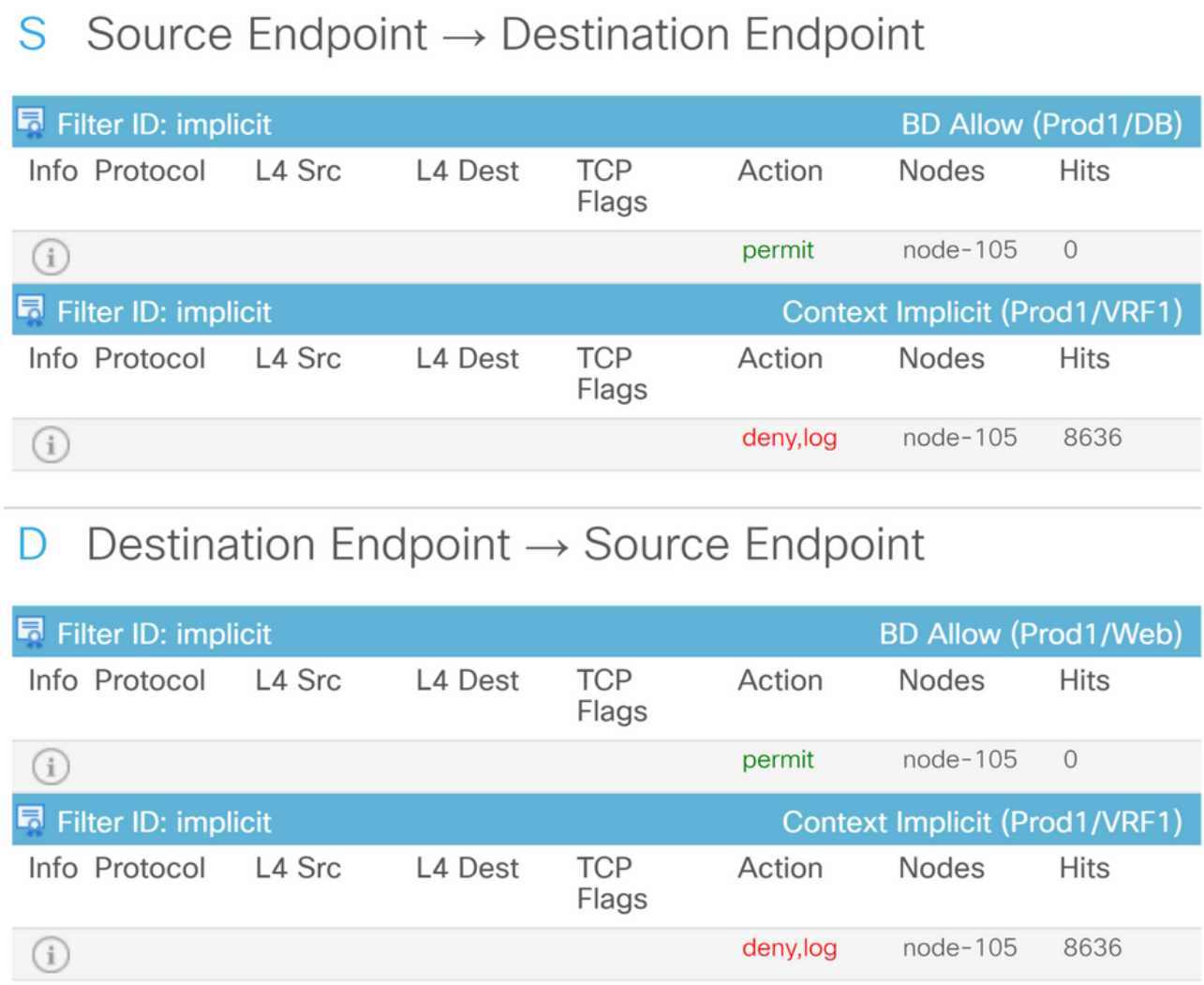
이러한 삭제 중 상당수는 예상된 동작이며 무시할 수 있습니다.

삭제 세부 정보



스위치 다이어그램의 노란색 'Packets dropped' 버튼을 사용하여 세부사항을 삭제하도록 드릴다운하면 삭제된 플로우에 대한 세부사항을 볼 수 있습니다.

계약 세부사항



Contracts(계약) 하위 메뉴로 이동하면 EPG 간에 정책 드롭오프를 일으키는 계약을 식별할 수 있습니다. 이 예에서는 일부 적중을 표시하는 Prod1/VRF1을 거부하는 것이 암시적입니다. 이것은 지정된 흐름(192.168.21.11 및 192.168.23.11)이 암시적 거부를 때리는 것을 반드시 의미하지는 않습니다. Hits of Context Implicit Deny 규칙이 증가하고 있으면 Prod1/DB와 Prod1/Web 간에 어떤 계약에도 도달하지 않은 트래픽이 있음을 의미합니다. 따라서 Implicit deny에 의해 삭제됩니다.

Application Profile Topology(애플리케이션 프로필 토폴로지) 보기의 Tenant(테넌트) > Topology(토폴로지) 왼쪽에서 Application Profile name(애플리케이션 프로필 이름)을 선택하면 어떤 계약이 DB EPG에 적용되는지 확인할 수 있습니다. 이 경우 EPG에 계약이 할당되지 않습니다.

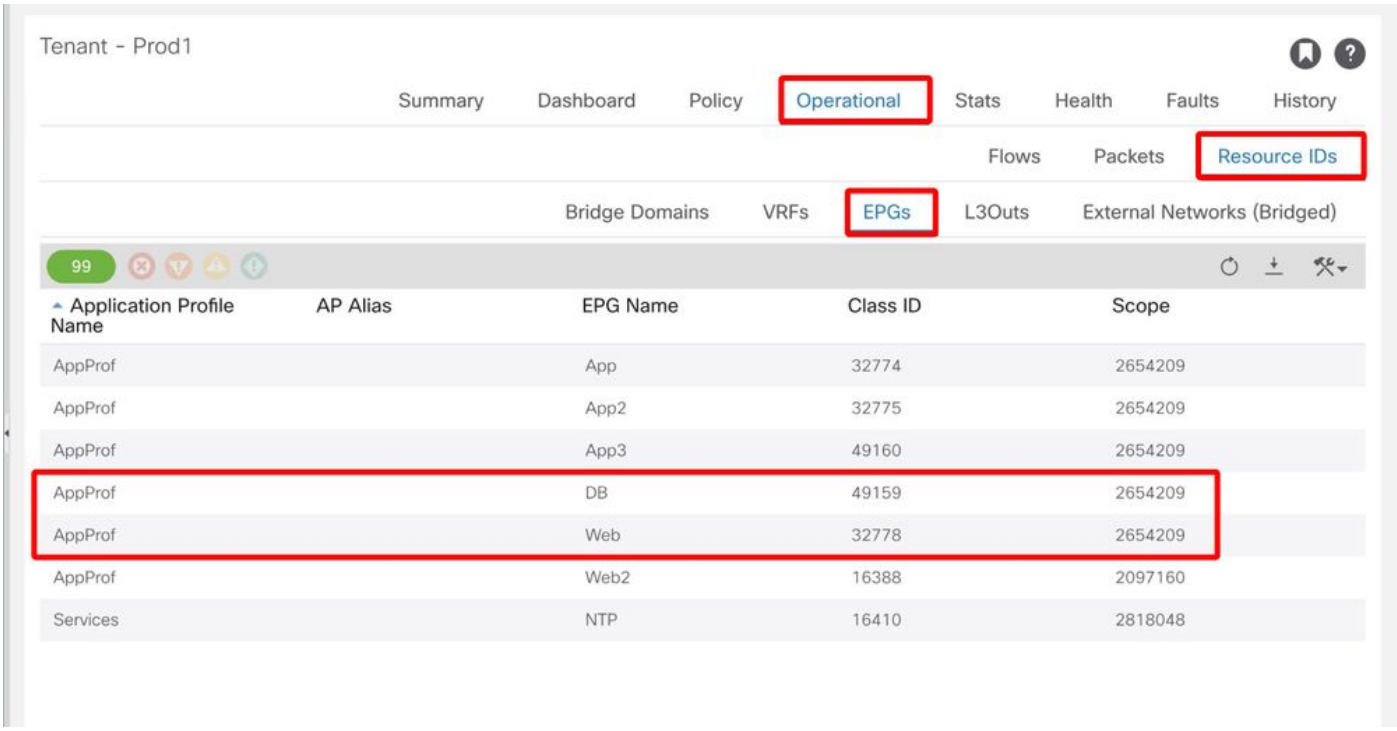
계약 시각화

이제 소스 및 대상 EPG가 알려졌으므로 다음과 같은 다른 관련 정보를 식별할 수도 있습니다.

- 영향을 받는 엔드포인트의 src/dst **EPG** pcTag. pcTag는 zoning-rule을 사용하여 EPG를 식별하는 데 사용되는 클래스 ID입니다.
- 영향을 받는 **엔드포인트**의 src/dst **VRFVNID**(범위라고도 함)

Tenant(테넌트)를 열고 > 왼쪽에서 Tenant name(테넌트 이름)을 선택한 다음 Operational(운영) > Resource IDs(리소스 ID) > EPGs(EPG)를 선택하면 APIC GUI에서 클래스 ID 및 범위를 쉽게 검색할 수 있습니다

EPG pcTag 및 범위를 찾기 위한 테넌트 리소스 ID



이 경우 클래스 ID 및 범위는 다음과 같습니다.

- 웹 EPG pcTag 32778
- 웹 EPG 범위 2654209
- DB EPG pcTag 49159
- DB EPG 범위 2654209

트러블슈팅 중인 트래픽 흐름에 적용된 정책 확인

아이배시

ACI 리프에 삭제된 패킷을 확인하는 흥미로운 툴은 iBash 명령줄입니다. '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
```

이전 출력에 따르면 leaf 스위치에서 EP 192.168.23.11에서 192.168.21.11로 소싱된 수많은 ICMP 패킷이 삭제되었음을 알 수 있습니다.

contract_parser 툴은 엔드포인트가 연결된 VRF에 적용된 실제 정책을 확인하는 데 도움이 됩니다.

```
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]
```

이는 leaf에 프로그래밍된 zoning rule을 통해 스위치에서 시행하는 정책을 통해서도 확인할 수 있다
.

```
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)      |
+---------+--------+--------+----------+----------------+---------+----------+------------+-----
-----+-------------------------+
```

Visibility & Troubleshooting 툴, contract_parser 툴, zoning-rule에서 이미 볼 수 있듯이 출력에 따라 트러블슈팅에서 소스와 대상 EPG 간에 contract가 없음을 확인합니다. 삭제된 패킷이 암시적 거부 규칙(5155)과 일치한다고 가정하기 쉽다.

ELAM 캡처

ELAM 캡처는 포워딩 세부사항을 확인하는 데 사용되는 ASIC 레벨 보고서를 제공하며, 이는 삭제된 패킷의 경우 삭제 이유를 나타냅니다. 이 시나리오와 같이 정책 삭제의 원인이 될 경우 ELAM 캡처의 출력은 다음과 같습니다.

ELAM 캡처 설정에 대한 자세한 내용은 이 장에서 다루지 않습니다. "Intra-Fabric Forwarding" 장을 참조하십시오.

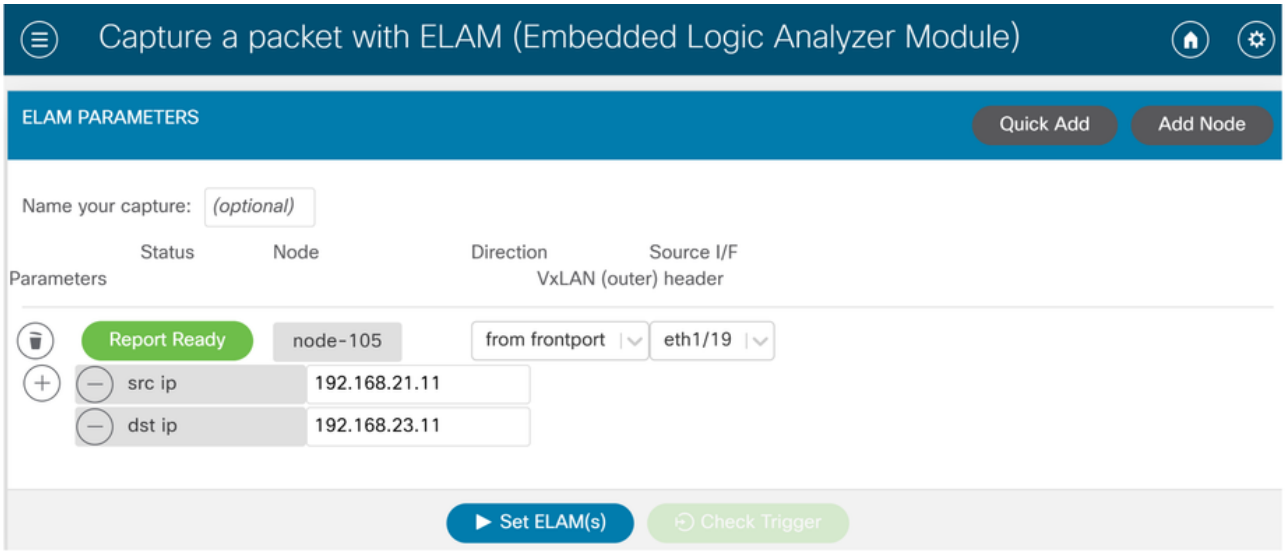
```
leaf5# vsh_lc
module-1# debug platform internal tah elam asic 0
module-1(DBG-elam)# trigger init in-select 6 out-select 0
module-1(DBG-elam)# trigger reset
module-1(DBG-elam-insel6)# set outer ipv4 src_ip 192.168.21.11 dst_ip 192.168.23.11
module-1(DBG-elam-insel6)# start
module-1(DBG-elam-insel6)#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

위의 ELAM 보고서는 정책 삭제로 인해 패킷이 삭제되었음을 명확히 보여줍니다.
'SECURITY_GROUP_DENY'

ELAM 보조자:

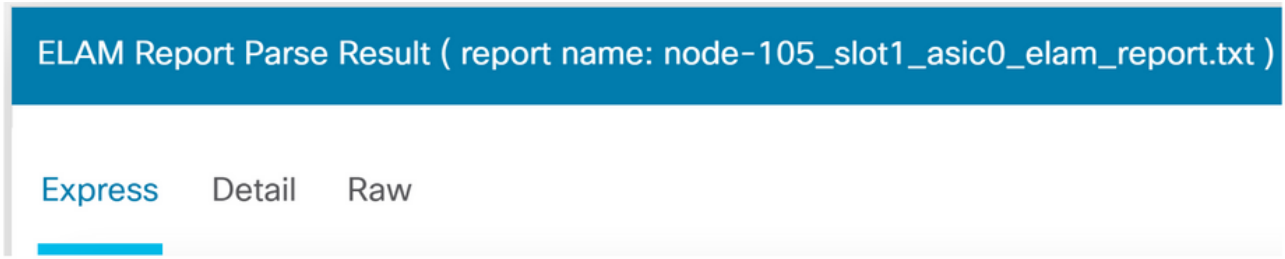
ELAM 캡처의 동일한 결과는 APIC GUI에서 ELAM Assistant App을 통해 표시할 수 있습니다.

설정



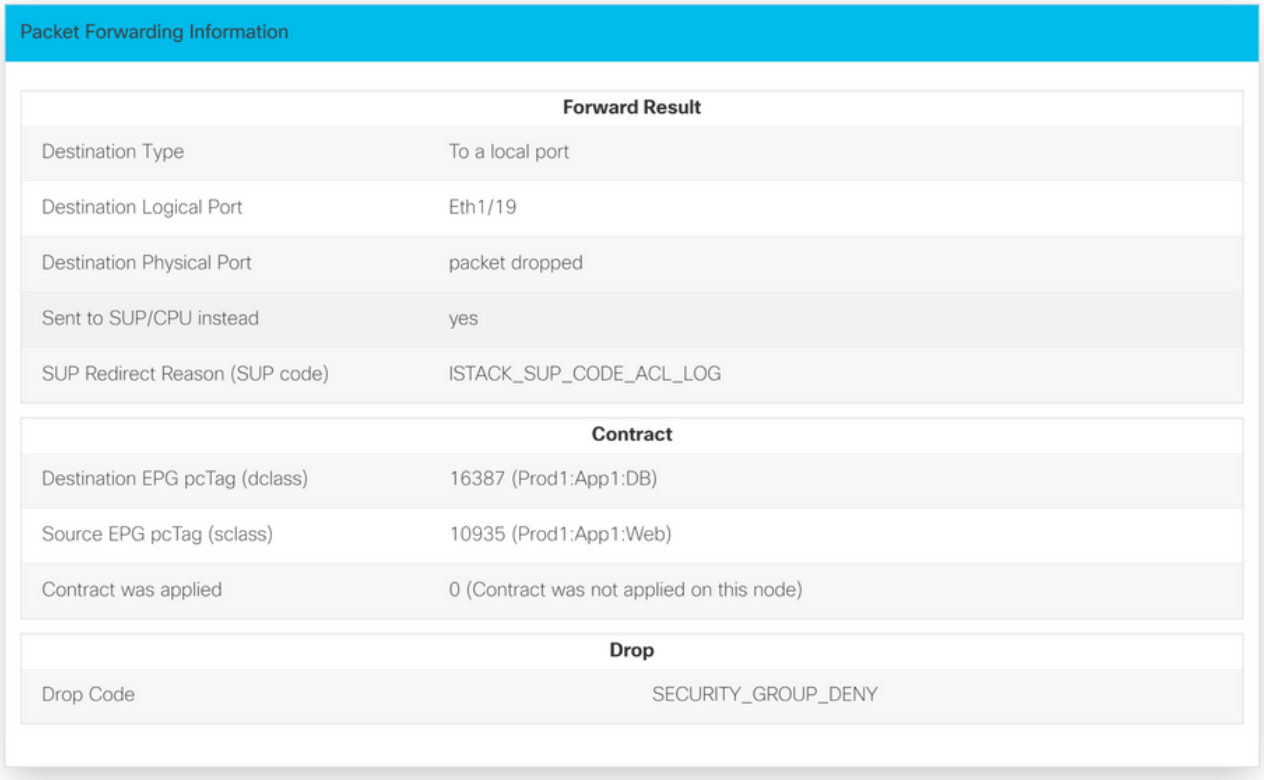
일반적으로 사용자는 관심 플로우에 대한 소스 및 대상 세부 정보를 모두 구성합니다. 이 예에서는 소스 EPG와 계약 관계가 없는 목적지 EPG의 엔드포인트로 향하는 트래픽을 캡처하는 데 src IP가 사용됩니다.

Elam Assistant Express 보고서



ELAM Assistant를 사용하여 볼 수 있는 3가지 출력 레벨이 있습니다. Express, Detail 및 Raw입니다.

Elam Assistant Express 보고서(계속)



Express Result(빠른 결과) 아래에서 Drop Code reason SECURITY_GROUP_DENY는 계약 적중의 결과임을 나타냅니다.

선호 그룹

계약 선호 그룹 정보

Contract Preferred Group이 구성된 VRF에서 EPG에 사용할 수 있는 정책 시행에는 두 가지 유형이 있습니다.

- 포함된 EPG: EPG는 계약 선호 그룹에 멤버십이 있는 경우 계약 없이 자유롭게 서로 통신할 수 있습니다. 이는 source-any-destination-any-permit 기본 규칙을 기반으로 합니다.
- 제외된 EPG: 선호 그룹의 멤버가 아닌 EPG는 계약을 통해 서로 통신해야 합니다. 그렇지 않으면 제외된 EPG와 모든 EPG 간의 거부 규칙이 적용됩니다.

Contract Preferred Group 기능은 VRF의 EPG 간 통신을 보다 효과적으로 제어할 수 있게 합니다. VRF의 EPG가 대부분 개방형 통신을 가져야 하지만 일부는 다른 EPG와의 통신이 제한적으로만 이루어져야 하는 경우, contract preferred group과 contract with filters의 조합을 구성하여 EPG 간 통신을 보다 정밀하게 제어합니다.

선호 그룹에서 제외된 EPG는 source-any-destination-any-deny 기본 규칙을 재정의하는 계약이 있는 경우에만 다른 EPG와 통신할 수 있습니다.

계약 선호 그룹 프로그래밍

기본적으로 계약 선호 그룹은 일반 계약의 반대입니다. 일반 계약의 경우 명시적 허용 영역 지정 규칙은 VRF 범위의 암시적 거부 영역 지정 규칙으로 프로그래밍됩니다. 선호 그룹의 경우 암시적 PERMIT zoning-rule은 가장 높은 숫자 우선순위 값으로 프로그래밍되고 특정 DENY zoning-rule은 선호 그룹 멤버가 아닌 EPG의 트래픽을 허용하지 않도록 프로그래밍됩니다. 그 결과, 거부 규칙이 먼저 평가되고 이러한 규칙과 플로우가 일치하지 않으면 해당 플로우가 암시적으로 허용됩니다.

기본 그룹 외부의 모든 EPG에는 항상 명시적 거부 영역 지정 규칙 쌍이 있습니다.

- 비선호 그룹 멤버에서 임의의 pcTag로의 하나(값 0).
- 임의의 pcTag(값 0)에서 비선호 그룹 멤버로의 또 다른 값

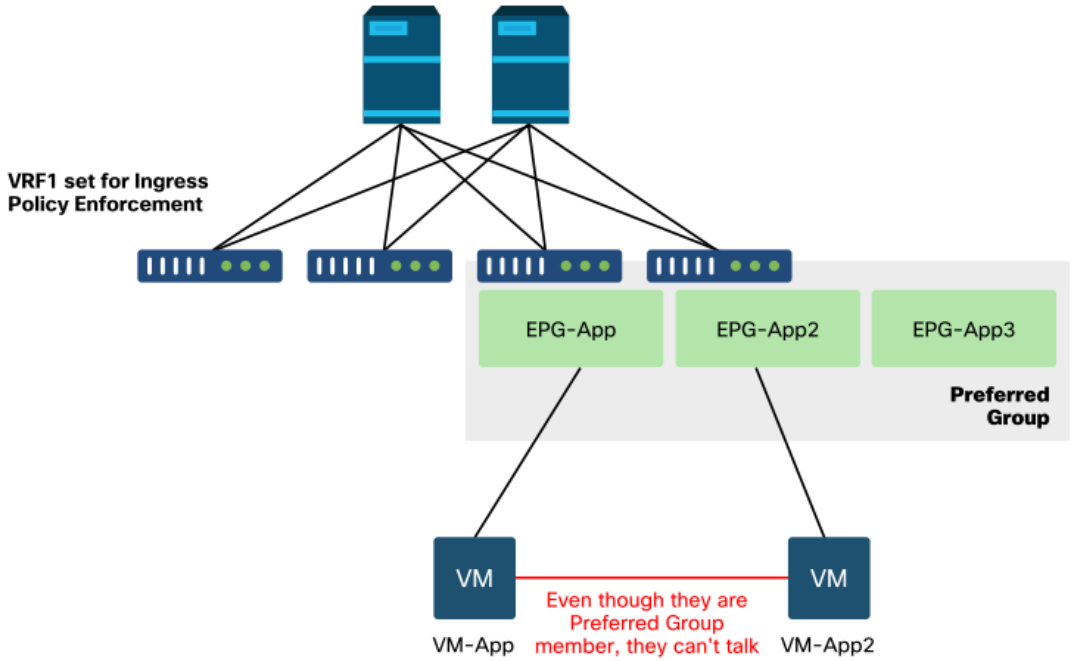
기본 그룹 문제 해결 시나리오

아래 그림에는 EPG App, App2 및 App3가 모두 Preferred Group Member로 구성된 논리적 토폴로지가 나와 있습니다.

VM-App은 EPG-App의 일부이고 VM-App2는 EPG-App2의 일부입니다. App 및 App2 EPG는 모두 기본 설정의 일부이므로 자유롭게 통신해야 합니다.

VM-App은 TCP 포트 6000에서 VM-App2로의 트래픽 흐름을 시작합니다. EPG-App 및 EPG-App2는 모두 VRF1의 일부로서 기본 그룹 멤버입니다. VM-App2는 TCP 포트 6000에서 패킷을 수신하지 않습니다.

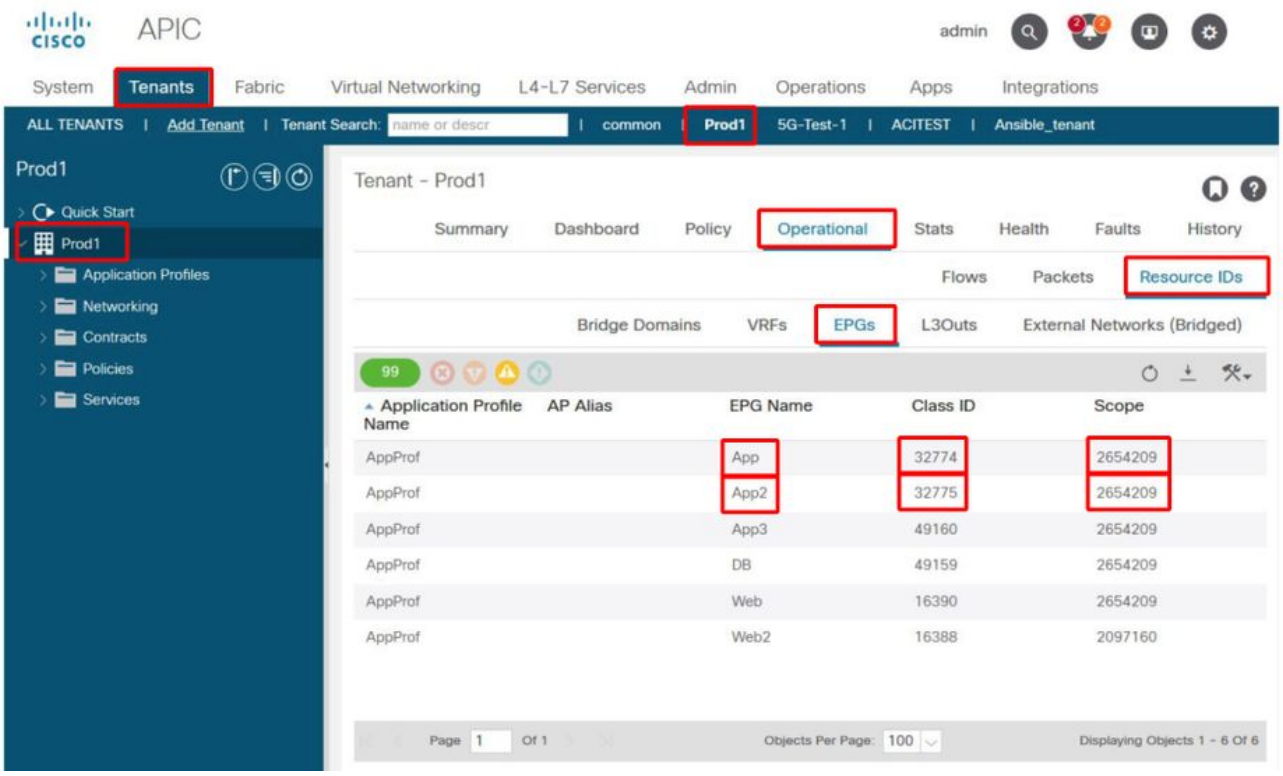
토폴로지



워크플로

1. EPG 앱의 pcTag 및 해당 VRF VNID/범위를 조회합니다.

EPG 및 VRF pcTag



2. 인그레스 리프에서 contract_parser.py를 사용하여 계약 프로그래밍을 확인합니다.

contract_parser.py 및/또는 'show zoning-rule' 명령을 사용하고 VRF를 지정합니다

```
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]
```

위의 출력을 살펴보면 우선 순위가 가장 높은 20의 암시적 허용 항목(ruleId 4165)이 관찰됩니다. 이 암시적 허용 규칙은 우선 순위가 더 낮고 트래픽 흐름을 허용하지 않는 명시적 거부 규칙이 없는 한 모든 트래픽 흐름을 허용합니다.

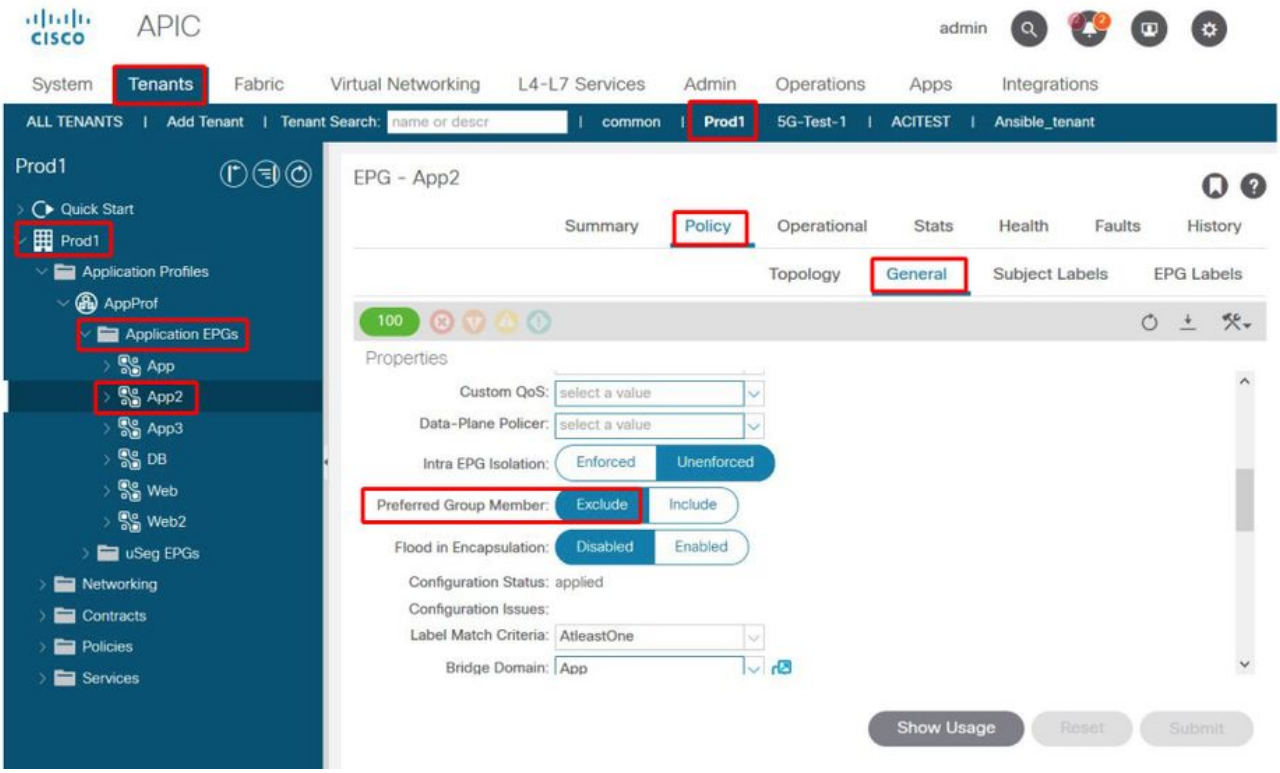
또한 EPG App2의 pcTag인 pcTag 32775에 대해 2개의 명시적 거부 규칙이 관찰됩니다. 이 2개의 명시적 거부 영역 지정 규칙은 EPG에서 EPG App2로 또는 그 반대로 트래픽을 허용하지 않습니다. 이러한 규칙에는 우선순위 18 및 19가 있으므로 기본 허용 규칙에 우선합니다.

결론적으로 EPG App2는 명시적 거부 규칙이 관찰되므로 선호 그룹 멤버가 아닙니다.

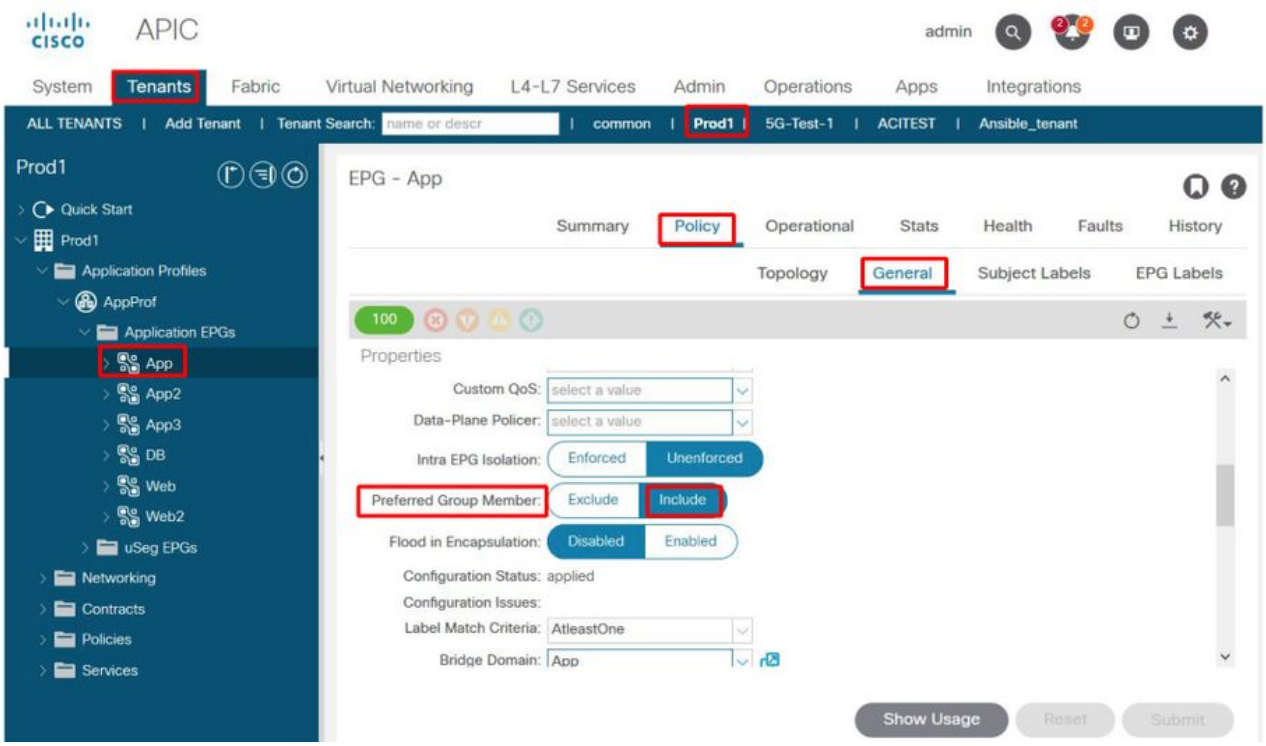
3. EPG 선호 그룹 구성원 구성 확인

APIC GUI를 탐색하고 EPG App2 및 EPG App Preferred Group Member Configuration(EPG 앱 선호 그룹 멤버 컨피그레이션)을 확인합니다. 다음 그림에서 EPG App2는 선호 그룹 멤버로 구성되지 않음을 확인합니다.

EPG App2 - 기본 그룹 구성원 설정이 제외됨



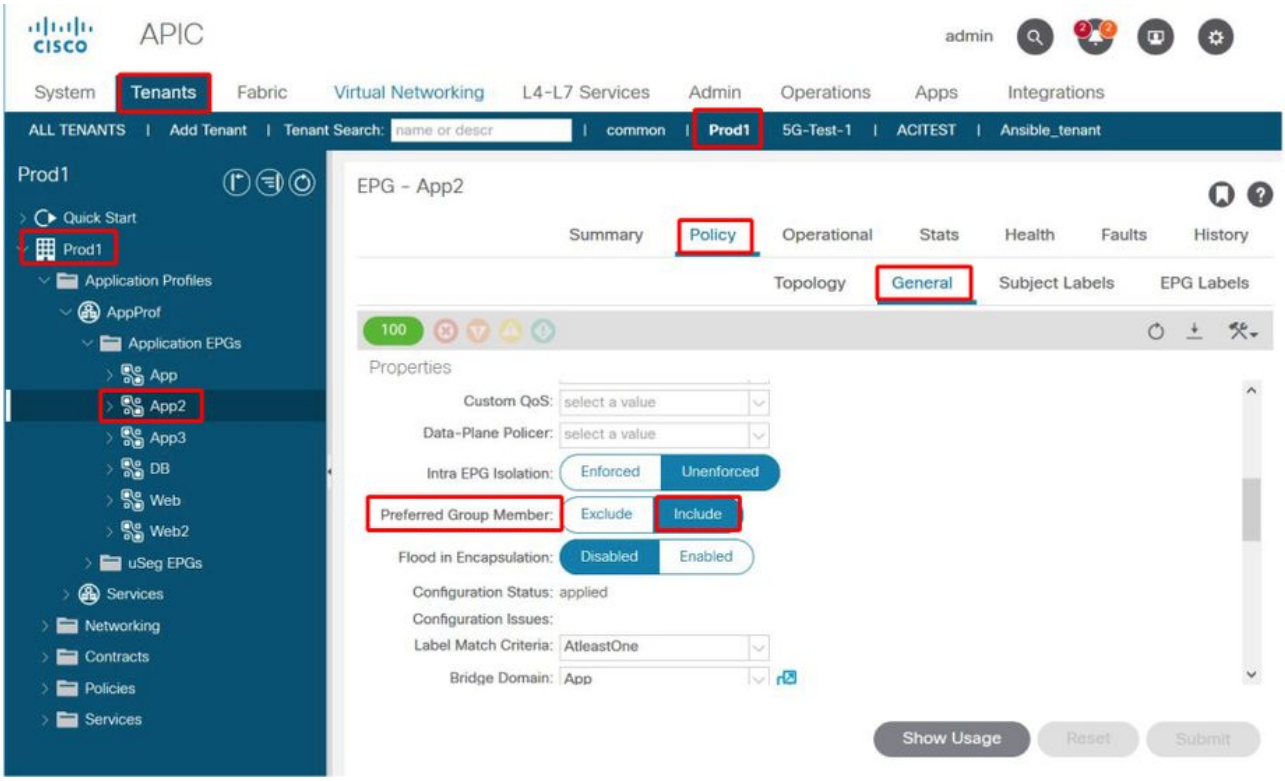
EPG 앱 - 기본 그룹 구성원 설정 포함



4. EPG App2를 선호 그룹 멤버로 설정

App2 EPG의 컨피그레이션을 변경하면 선호 그룹이 선호 그룹의 일부로서 자유롭게 통신할 수 있습니다.

EPG App2 - 기본 그룹 구성원 설정 포함



5. src EP가 있는 leaf에서 contract_parser.py를 사용하여 계약 프로그래밍을 다시 확인합니다.

contract_parser.py를 다시 사용하고 VRF 이름을 지정하여 EPG App2에 대한 명시적 거부 규칙이 제거되었는지 확인합니다.

```
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]
```

EPG App2 및 해당 pcTag 32775에 대한 명시적 거부 규칙은 위의 출력에서 더 이상 관찰되지 않습니다. 이는 EPG 앱과 EPG 앱2의 EP 간의 트래픽이 이제 가장 높은 우선순위 20의 암시적 허용 규칙(ruleId 4165)과 일치함을 의미합니다.

vzAny에서 EPG로

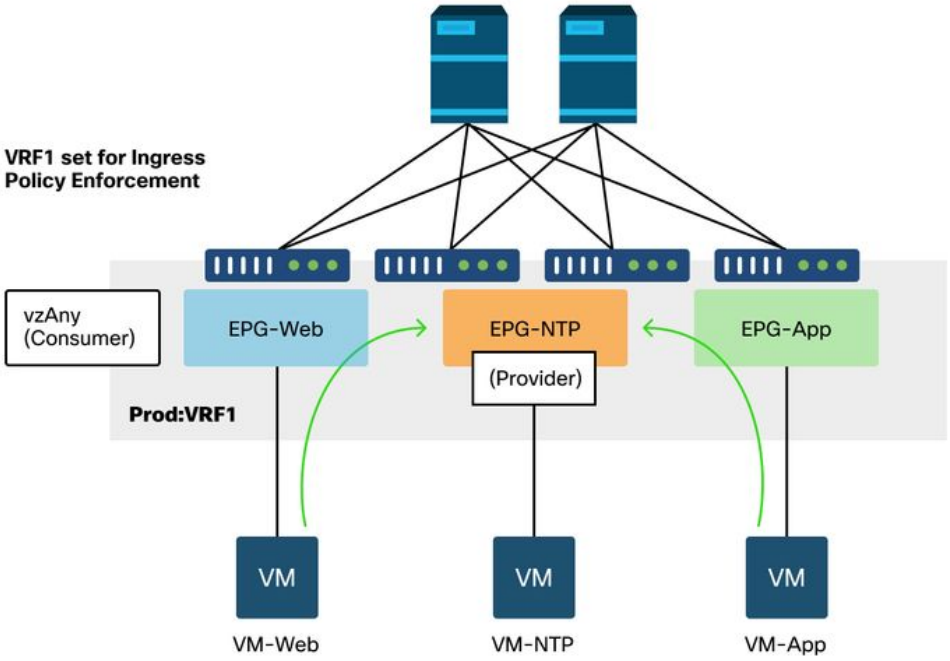
vzAny 정보

하나 이상의 EPG 간에 contract를 구성할 때 contract를 소비된 관계 또는 제공된 관계로 구성할 수 있습니다. EPG의 수가 증가하면 EPG 간의 계약 관계도 증가할 수 있습니다. 일부 일반적인 활용 사례에서는 트래픽 흐름을 다른 특정 EPG와 교환하려면 모든 EPG가 필요합니다. 이러한 활용 사례는 동일한 VRF(예: NTP 또는 DNS) 내의 다른 모든 EPG에서 사용해야 하는 서비스를 제공하는 EP를 포함하는 EPG일 수 있습니다. vzAny를 사용하면 모든 EPG와 특정 EPG 간의 계약 관계를 구성할 때 운영 오버헤드가 줄어들어 다른 모든 EPG에서 서비스를 사용할 수 있습니다. 또한 vzAny 계약 관계마다 2개의 zoning-rule만 추가되므로 vzAny를 통해 리프 스위치에서 훨씬 더 효율적인 보안 정책 CAM을 사용할 수 있습니다.

활용 사례

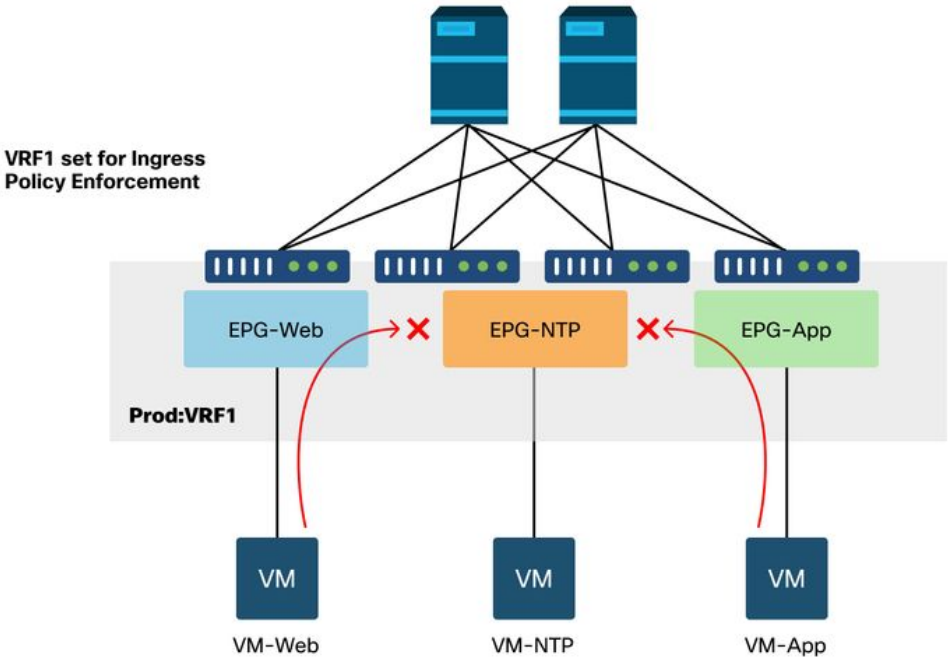
아래 그림에서는 EPG 웹 및 앱의 VM-Web 및 VM-App이 각각 EPG-NTP의 VM-NTP에서 NTP 서비스를 사용해야 하는 활용 사례를 설명합니다. EPG NTP에서 제공된 계약을 구성하고, 그런 다음 EPG 웹 및 앱에서 소비된 계약과 동일한 계약을 갖는 대신, vzAny는 VRF Prod:VRF1의 각 EPG가 EPG NTP의 NTP 서비스를 소비하도록 허용합니다.

vzAny — VRF Prod의 모든 EPG:VRF1은 EPG NTP의 NTP 서비스를 사용할 수 있습니다.



NTP 서비스를 사용하는 EPG 간에 contract가 없을 경우 EPG 간에 드롭이 관찰되는 시나리오를 가정해 보십시오.

문제 해결 시나리오 - 계약이 없을 경우 트래픽 중단

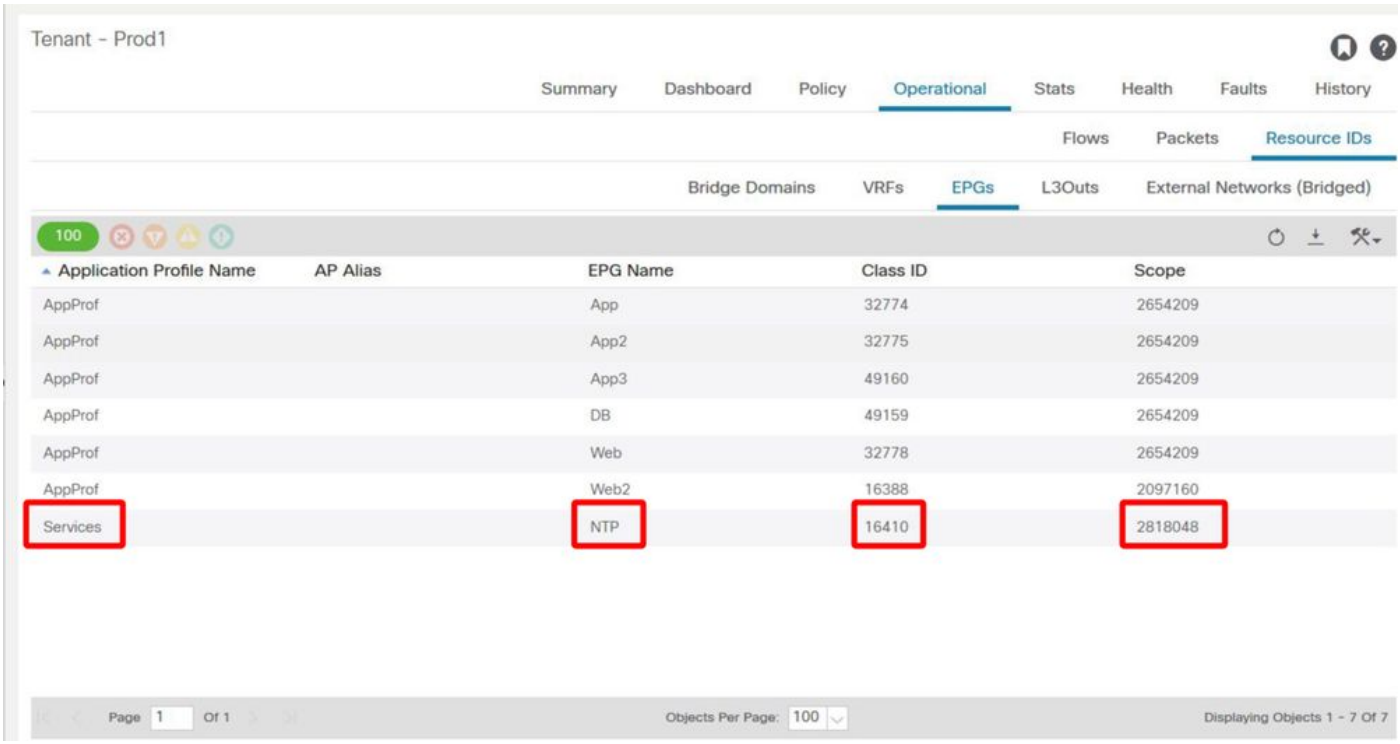


워크플로

1. EPG NTP 및 해당 VRF VNID/범위의 pcTag를 조회합니다.

'Tenant(테넌트) > Operational(운영) > Resource IDs(리소스 ID) > EPGs'에서는 pcTag 및 범위를 찾을 수 있습니다.

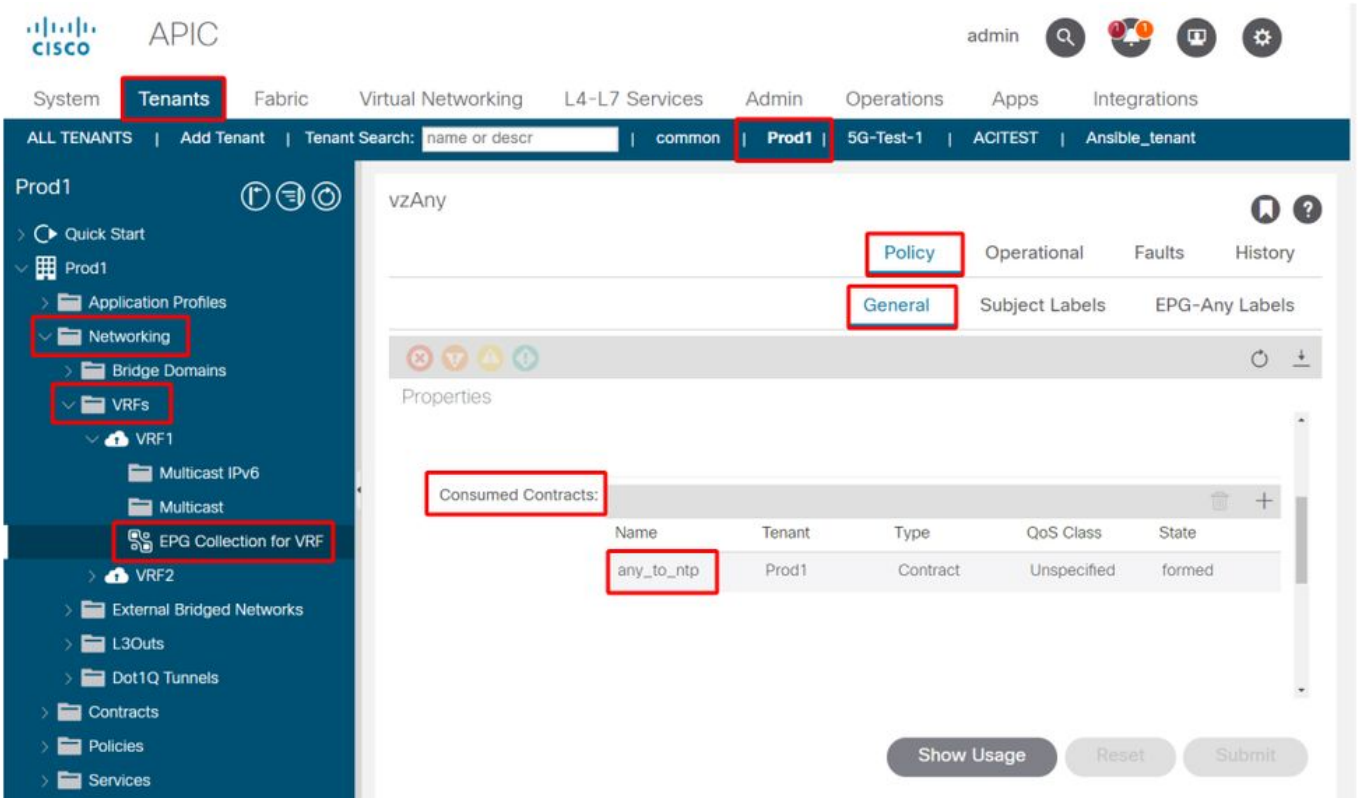
EPG NTP pcTag 및 해당 VRF VNID/범위



2. 계약이 vzAny 소비된 계약으로 VRF의 일부로 구성되어 있는지 확인합니다.

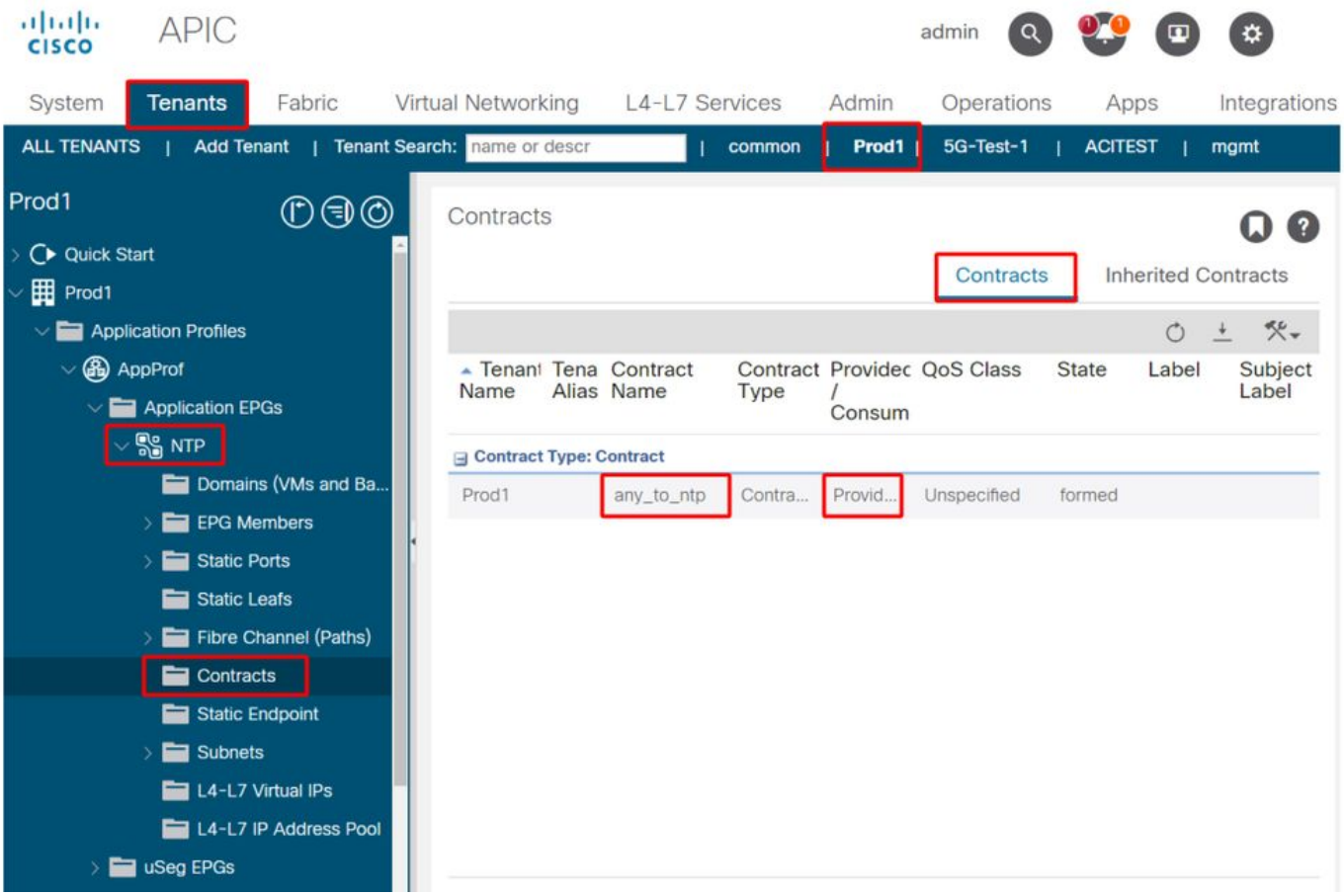
VRF로 이동하여 'VRF용 EPG 컬렉션' 아래에 vzAny로 구성된 소비 계약이 있는지 확인합니다.

Contract가 소비된 vzVRF에서 Any contract로 구성됨



3. EPG NTP에서 동일한 계약이 제공된 계약으로 적용되는지 확인합니다.

계약 관계를 설정하려면 VRF의 다른 EPG에 NTP 서비스를 제공하는 EPG NTP에 제공된 계약으로 동일한 계약을 적용해야 합니다.



4. contract_parser.py 또는 'show zoning-rule'을 사용하여 인그레스 리프에 대한 zoning-rule 확인

모든 EPG와 EPG NTP 간의 양방향 트래픽 흐름을 허용하려면 인그레스 리프에 2개의 조닝 규칙이 있어야 합니다(계약 주체가 양방향을 허용하도록 설정된 경우). 조닝 규칙 프로그래밍에서 'Any EPG'는 pcTag 0으로 표시됩니다.

VRF를 지정하는 동안 인그레스 리프에서 contract_parser.py 또는 'show zoning-rule' 명령을 사용하면 zoning-rule이 프로그래밍되도록 할 수 있습니다.

VRF에 있는 다른 EPG에서 EPG NTP로/로부터의 트래픽을 허용하는 조닝 규칙

contract_parser.py 및 'show zoning-rule'을 사용하여 vzAny 기반 zoning-rule이 있는지 확인합니다.

여기에서는 두 가지 유형의 규칙이 분명합니다.

1. Any를 NTP에 허용하고 그 반대의 경우도 허용하는 규칙 4156 및 규칙 4168. 우선 순위 13 및 14: 모든 EPG(pcTag 0)에서 EPG NTP(pcTag 49161)로의 트래픽 흐름을 허용하는 조닝 규칙 EPG NTP(pcTag 46161)에서 다른 EPG(pcTag 0)로의 트래픽 흐름을 허용하는 조닝 규칙.
2. 규칙 4165 - 우선순위 21의 any to any 규칙(기본값).

우선순위가 가장 낮으면 VRF의 모든 EPG가 NTP EPG에 액세스합니다.

```
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)  |
+---------+--------+--------+----------+---------+---------+---------+------------+----------+--
```

-------------------+

EPG에 공유된 L3Out

공유 L3Out 정보

공유 레이어 3 OUT은 일부 서비스(외부 액세스)를 제공하는 하나의 VRF에 L3Out을 설정하고 하나 이상의 다른 VRF가 이 L3Out을 사용하도록 하는 구성입니다. 공유 L3Out에 대한 자세한 내용은 "외부 라우팅" 장에서 확인할 수 있습니다.

공유 L3Out을 수행할 때는 계약의 공급자가 공유 L3Out이고 EPG가 계약의 소비자가 되는 것이 좋습니다. 이 시나리오에 대해서는 이 섹션에서 설명합니다.

이와 반대로 L3Out이 EPG에서 제공하는 서비스를 소비하는 것은 권장되지 않습니다. 그 이유는 공유 서비스의 경우 영역 지정 규칙이 소비자 VRF에만 설치되기 때문에 확장성과 관련이 있습니다. 소비 및 제공의 원칙은 트래픽 흐름이 시작되는 위치를 나타냅니다. 기본적으로 인그레스(ingress) 정책이 적용되므로 소비자 측에서, 더 구체적으로는 인그레스 리프(비경계 리프)에서 정책이 적용됩니다. 인그레스 리프가 정책을 시행하려면 대상의 pcTag가 필요합니다. 이 시나리오에서 대상은 외부 EPG pcTag입니다. 인그레스 리프는 정책 시행을 수행하고 패킷을 경계 리프로 전달합니다. 보더 리프는 LPM(Route Lookup)을 수행하는 패브릭 링크에서 패킷을 수신하고 목적지 접두사의 인접지로 패킷을 전달합니다.

그러나 보더 리프는 목적지 EP로 트래픽을 전송할 때 어떤 정책 시행도 수행하지 않으며 반환 트래픽 플로우에서 소스 EP로 다시 이동하지도 않습니다.

그 결과 인그레스 비 BL 리프의 Policy CAM만 (소비자 VRF에) 항목이 설치되어 있고 BL의 Policy CAM은 영향을 받지 않습니다.

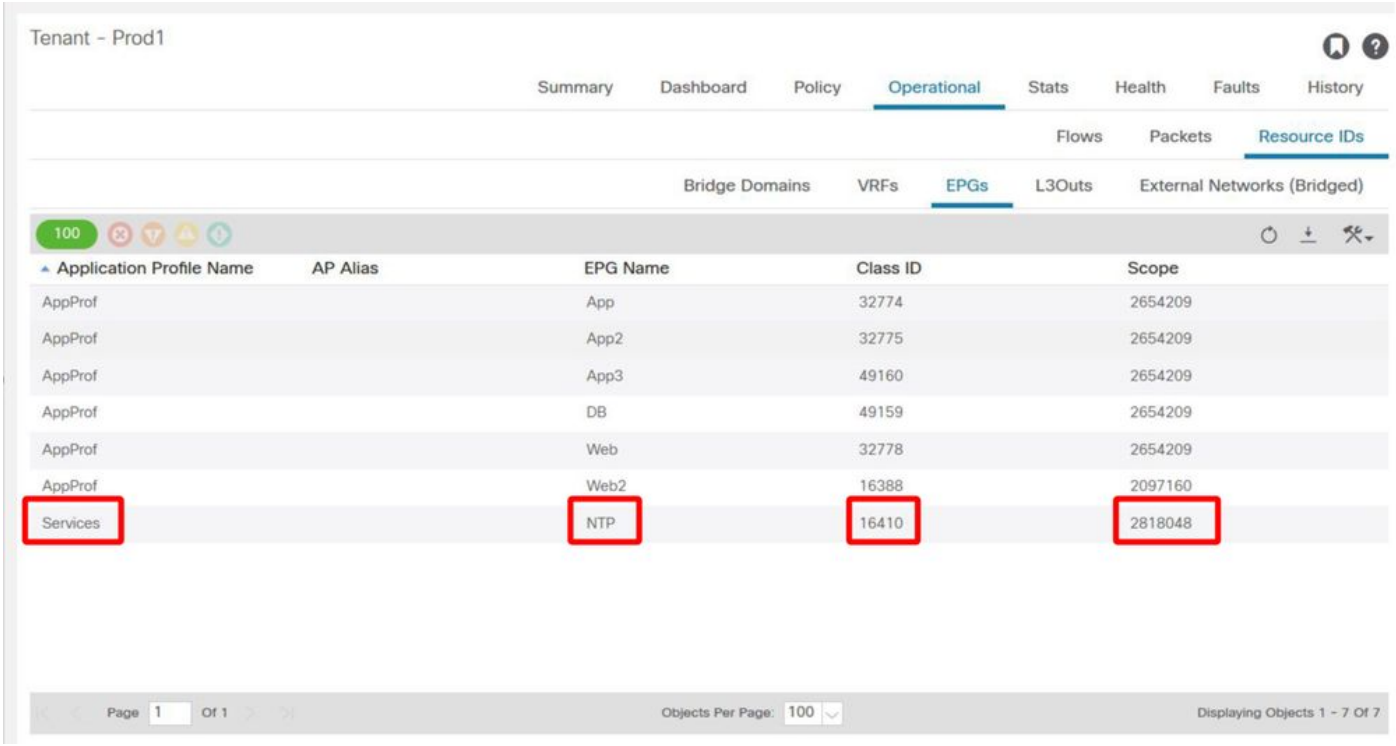
공유 L3out 문제 해결

워크플로

1. 소비자 EPG의 EPG pcTag 및 VRF VNID/범위 확인

공유 L3Out에서는 zoning-rule이 소비자 VRF에만 설치됩니다. 공급자는 모든 소비자 VRF에서 이 pcTag를 사용할 수 있는 글로벌 pcTag(16k 미만)를 보유해야 합니다. 이 시나리오에서 제공자는 외부 EPG이며 전역 pcTag를 보유합니다. 소비자 EPG에는 평소와 같이 로컬 pcTag가 있습니다.

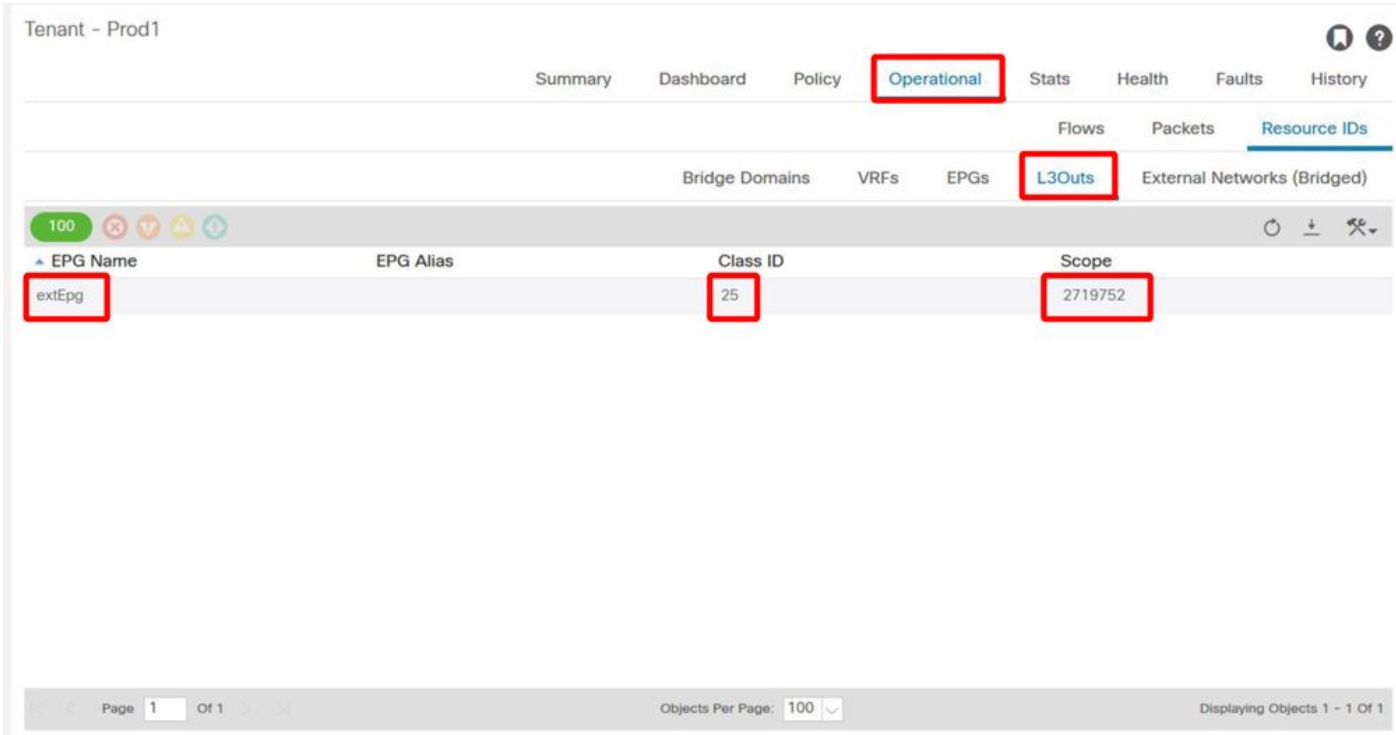
소비자 EPG의 pcTag



2. 공급자 L3Out EPG의 pcTag 및 VRF VNID/범위 확인

1단계에서 언급한 바와 같이, 사업자 L3Out EPG에는 소비자 VRF로 유출되는 L3Out의 접두사로 전역 범위 pcTag가 있습니다. 따라서 L3Out EPG pcTag는 소비자 VRF의 pcTag와 중복되지 않아야 하므로 전역 pcTag 범위 내에 있습니다.

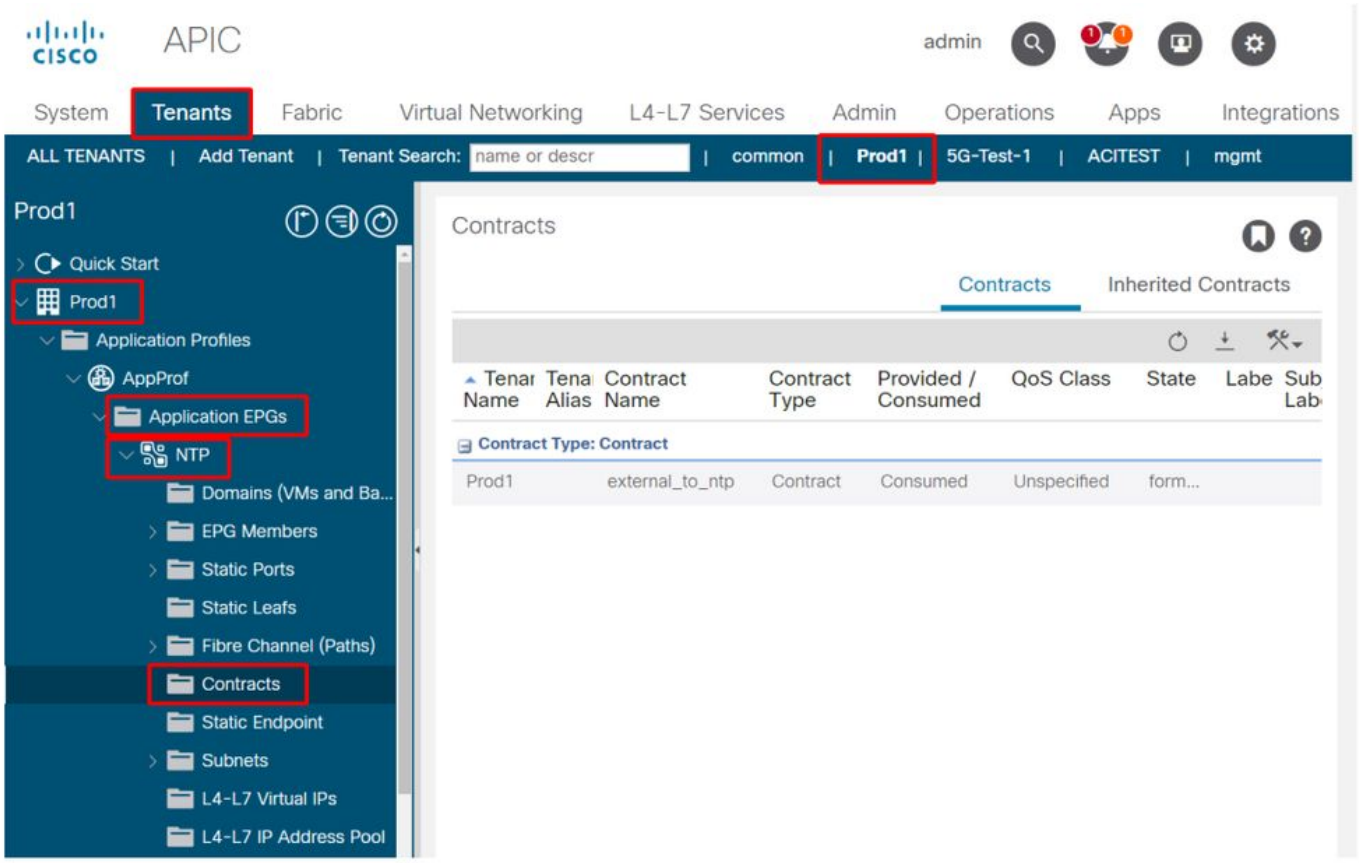
공급자 외부 EPG의 pcTag



3. 소비자 EPG에 가져온 테넌트 범위 계약 또는 글로벌 계약이 구성되어 있는지 확인합니다

EPG/BD에 서브넷이 정의되어 있는 소비자 EPG NTP가 '테넌트' 또는 '글로벌' 범위 계약을 소비합니다

EPG에서 소비한 계약



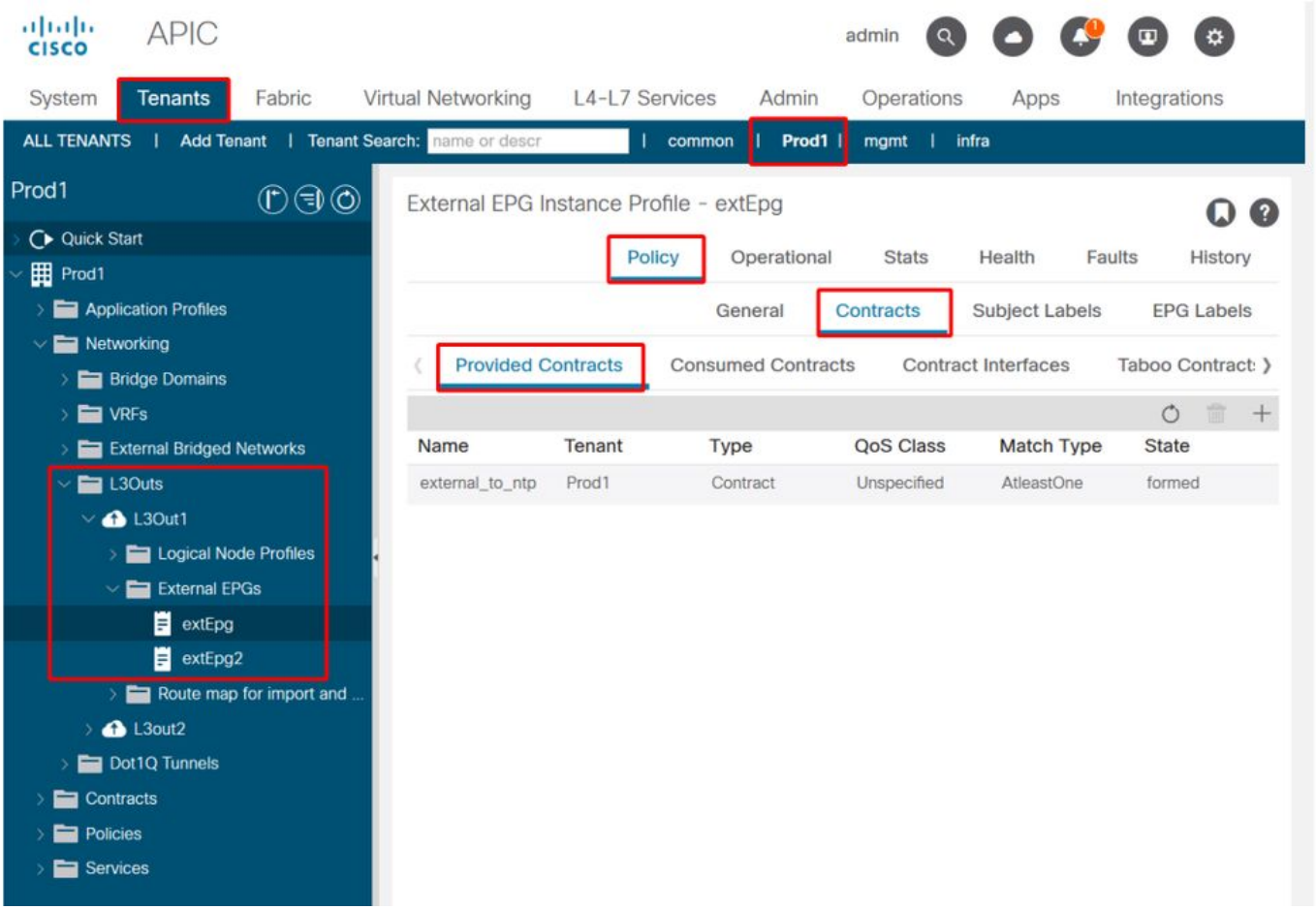
4. 소비자 EPG의 BD에 범위가 'VRF 간 공유'로 설정된 서브넷이 있는지 확인합니다.

EPG의 서브넷은 브리지 도메인 아래에 구성되지만 'shared between VRF' 플래그(라우티드 누설을 허용)와 'advertised externally' 플래그(L3Out에 광고를 허용)가 있어야 합니다

5. 공급자 L3Out EPG에 가져온 테넌트 범위 계약 또는 전역 계약이 구성되어 있는지 확인합니다

L3Out EPG에는 테넌트 범위 계약 또는 글로벌 계약이 제공된 계약으로 구성되어 있어야 합니다.

공급자 L3Out의 계약



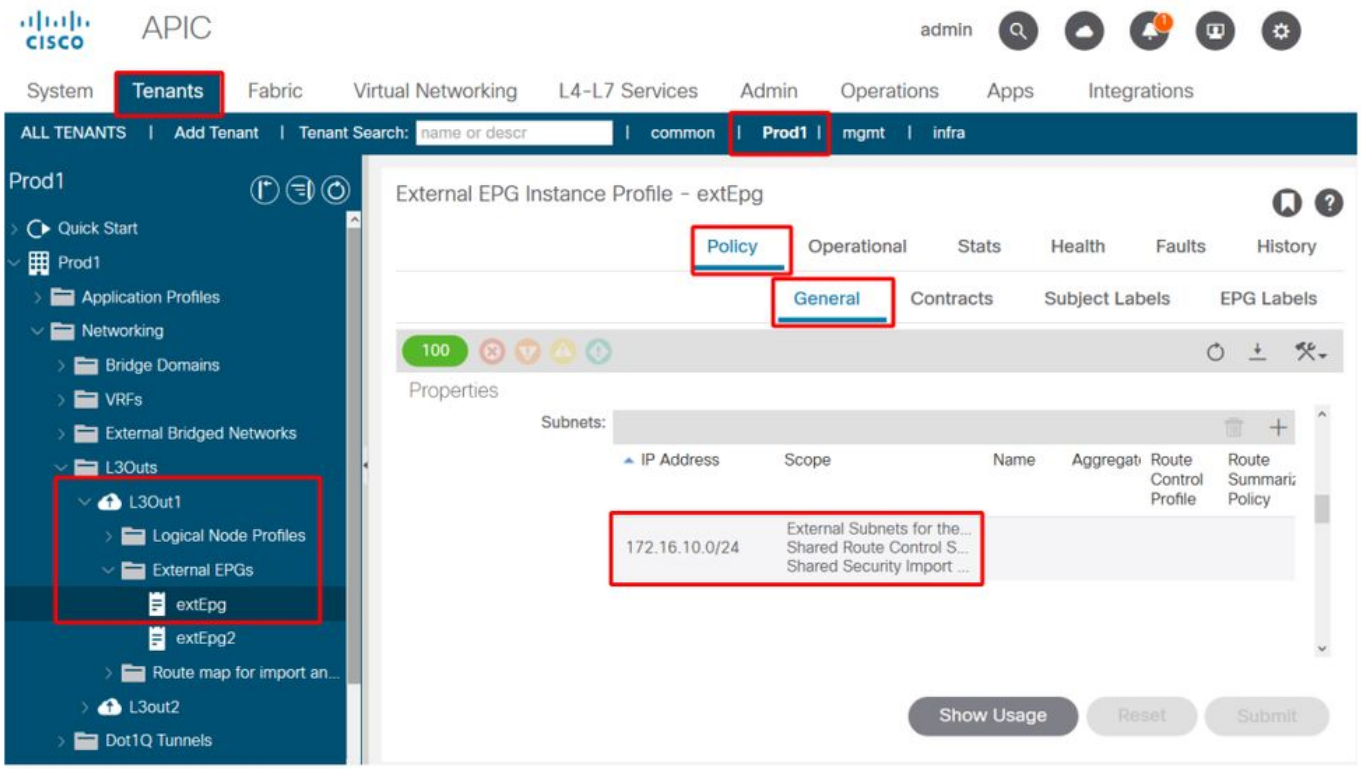
6. 공급자 L3Out EPG에 필요한 범위가 선택된 서브넷이 있는지 확인합니다

공급자 L3Out EPG에는 다음 범위로 유출될 접두사가 구성되어 있어야 합니다.

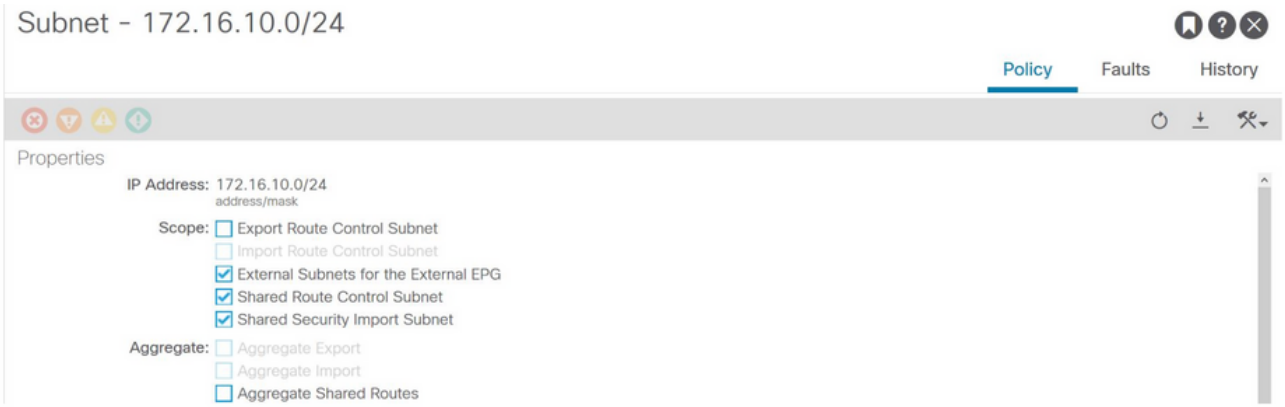
- 외부 EPG에 대한 외부 서브넷.
- 공유 경로 제어 서브넷
- 공유 보안 가져오기 서브넷.

L3Out EPG의 서브넷 플래그에 대한 자세한 내용은 "외부 전달" 장을 참조하십시오.

외부 EPG 서브넷 설정



외부 EPG 서브넷 설정이 확장됨



7. 소비자 VRF에 대한 비 BL에서 L3Out EPG 서브넷의 pcTag를 확인합니다

외부 EPG 서브넷으로 향하는 트래픽이 비 BL을 인그레스할 경우, 목적지 접두사에 대해 조회가 수행되어 pcTag를 확인합니다. 이는 non-BL에서 다음 명령어를 사용하여 확인할 수 있다.

이 출력은 소비자 VRF VNID인 VNI 2818048의 범위에서 사용됩니다. 테이블을 보면 동일한 VRF에 있지 않더라도 소비자가 목적지의 pcTag를 찾을 수 있습니다.

```
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
```

위의 출력은 L3Out EPG 서브넷과 전역 pcTag 25의 조합을 보여줍니다.

8. 소비자 VRF에 대한 non-BL에서 프로그램된 zoning-rule을 확인합니다

'contract_parser.py' 또는 'show zoning-rule' 명령을 사용하고 VRF를 지정합니다.

아래의 명령 출력에는 소비자 EPG 로컬 pcTag 16410에서 L3Out EPG 글로벌 pcTag 25로의 트래픽을 허용하는 두 개의 조닝 규칙이 설치됩니다. 이는 소비자 VRF의 범위인 2818048 범위에 있습니다.

```
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. 사업자 VRF에 대한 BL의 프로그램된 zoning-rule을 확인합니다.

'contract_parser.py' 또는 'show zoning-rule' 명령을 사용하고 VRF를 지정합니다. 다음 명령 출력은 이전에 여러 번 **설명했듯이** 사업자 VRF에 특정 zoning-rule이 없음을 보여줍니다.

사업자 VRF의 2719752이 되는 범위 내에 있다.

```
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]
```

이 번역에 관하여

Cisco는 전 세계 사용자에게 다양한 언어로 지원 콘텐츠를 제공하기 위해 기계 번역 기술과 수작업 번역을 병행하여 이 문서를 번역했습니다. 아무리 품질이 높은 기계 번역이라도 전문 번역가의 번역 결과물만큼 정확하지는 않습니다. Cisco Systems, Inc.는 이 같은 번역에 대해 어떠한 책임도 지지 않으며 항상 원본 영문 문서(링크 제공됨)를 참조할 것을 권장합니다.