Solução de problemas de políticas de segurança da ACI - Contratos

Contents

Introduction

Este documento descreve as etapas para entender e solucionar problemas das políticas de segurança da ACI, conhecidas como Contratos.

Informações de Apoio

O material deste documento foi extraído do livro Troubleshooting Cisco Application Centric Infrastructure, Second Edition, especificamente as Security Policies - Overview, Security Policies - Tools, Security Policies - EPG to EPG, Security Policies - Preferred group e Security Policies - vzAny to EPG Chapter.

Overview

A arquitetura de segurança fundamental da solução ACI segue um modelo de permissão. A menos que um VRF seja configurado no modo **não imposto**, todos os fluxos de tráfego de EPG para EPG são implicitamente descartados. Como implícito no modelo de lista de permissões pronto para uso, a configuração VRF padrão está no modo **imposto**. Os fluxos de tráfego podem ser permitidos ou explicitamente negados pela implementação de regras de zoneamento nos nós do switch. Essas regras de zoneamento podem ser programadas em uma variedade de configurações diferentes, dependendo do fluxo de comunicação desejado entre grupos de

endpoint (EPG) e do método usado para defini-los. Observe que as entradas de regra de zoneamento não são stateful e geralmente permitirão/negarão com base em porta/soquete determinados dois EPGs, uma vez que a regra foi programada.

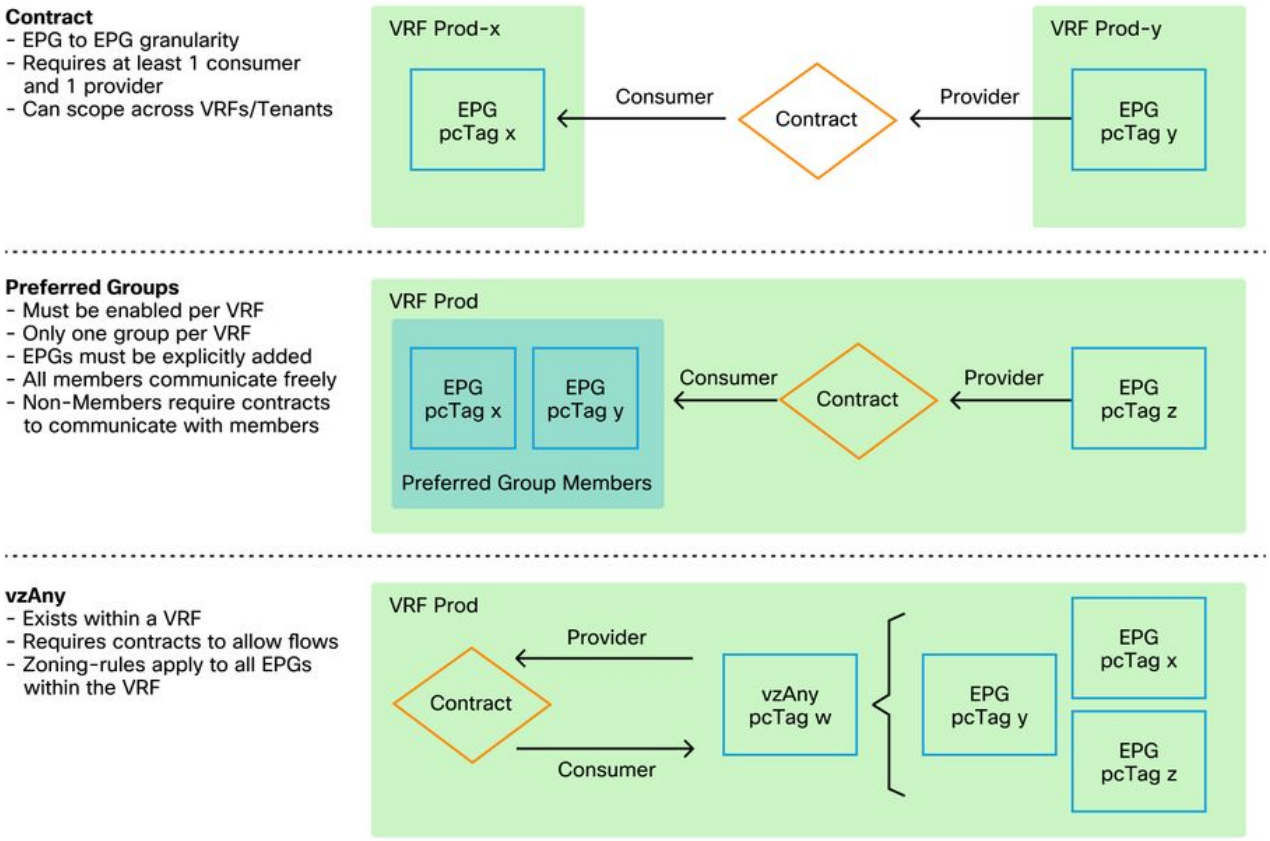
Métodos para programar regras de zoneamento

Os principais métodos para programar regras de zoneamento na ACI são:

- **Contratos EPG-para-EPG:** geralmente exige pelo menos um consumidor e um provedor para programar regras de zoneamento em dois ou mais grupos de endpoint distintos.
- **Grupos preferidos:** requer a ativação do agrupamento no nível VRF; somente um grupo pode existir por VRF. Todos os membros do grupo podem se comunicar livremente. Os não membros exigem contratos para permitir fluxos para o grupo preferido.
- **vzAny:** Uma 'coleção de EPG' que é definida em um determinado VRF. vzAny representa todos os EPGs no VRF. O uso de vzAny permite fluxos entre um EPG e todos os EPGs dentro do VRF através de uma conexão de contrato.

O diagrama a seguir pode ser usado para fazer referência à granularidade da regra de zoneamento que cada um dos métodos acima permite controlar:

Comparação entre metodologias de regra de zoneamento



Embora utilize o método de contrato de programação de regras de zoneamento, há uma opção para definir o escopo do contrato. Esta opção deve ser considerada cuidadosamente se for necessário algum vazamento de rota/projeto de serviço compartilhado. Se o desejo é passar de um VRF para outro dentro da estrutura da ACI, os contratos são o método para fazer isso.

Os valores de escopo podem ser os seguintes:

- **Aplicação:** um relacionamento de consumidor/fornecedor do contrato só programará regras entre EPGs que são definidos no mesmo perfil de aplicação. A reutilização do mesmo contrato em outros EPGs do perfil de aplicativo não permitirá a diafonia entre eles.
- **VRF (padrão):** um relacionamento de consumidor/fornecedor do contrato programará regras entre EPGs que são definidos no mesmo VRF. A reutilização do mesmo contrato em outros EPGs do perfil de aplicativo permitirá a diafonia entre eles. Tome cuidado para garantir que somente os fluxos desejados sejam permitidos, caso contrário um novo contrato deve ser definido para evitar diafonia não intencional.
- **Locatário:** um relacionamento de consumidor/fornecedor contratual programará regras entre EPGs que são definidos dentro do mesmo locatário. Se houver EPGs vinculados a vários VRFs em um único locatário e eles consumirem/fornecerem o mesmo contrato, esse escopo poderá ser usado para induzir vazamento de rota para permitir a comunicação entre VRFs.
- **Global:** um relacionamento contratual entre o consumidor e o fornecedor programará regras entre EPGs em qualquer usuário em uma estrutura da ACI. Este é o escopo mais alto possível da definição, e deve-se tomar muito cuidado quando isso é habilitado em contratos previamente definidos para evitar vazamento de fluxo não intencional.

Lendo uma entrada de regra de zoneamento

Depois que a regra de zoneamento for programada, ela aparecerá como a seguinte em uma folha:

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

- **ID da regra:** a ID da entrada da regra. Nenhum significado real além de atuar como um identificador único.
- **EPG de origem:** um ID exclusivo por VRF (pcTag) do grupo de endpoint de origem.
- **Dst EPG:** um ID exclusivo por VRF (pcTag) do grupo de endpoint de destino.
- **FilterID:** a ID do filtro com o qual a regra está tentando fazer correspondência. O Filtro contém as informações de protocolo em relação às quais a regra corresponderá.
- **Dir:** a direcionalidade da regra de zoneamento.
- **OperSt:** o Estado operacional da regra.
- **Escopo:** um ID exclusivo do VRF ao qual a regra corresponderá.
- **Nome:** o nome do contrato que resultou na programação dessa entrada.
- **Ação:** o que a folha fará quando corresponder a essa entrada. Inclui: [Descartar, Permitir, Registrar, Redirecionar].
- **Prioridade:** a ordem na qual as regras de zoneamento serão validadas para ação dada uma correspondência de Escopo, SrcEPG, DstEPG e Entradas de Filtro.

CAM (Content-Addressable Memory) de políticas

Àmedida que cada regra de zoneamento é programada, uma matriz da entrada de regra de zoneamento mapeada em relação às entradas de filtro começará a consumir a **política CAM** nos

switches. Ao projetar os fluxos permitidos através de uma estrutura da ACI, deve-se ter cuidado especial ao reutilizar contratos, em vez de criar novos, dependendo do projeto final. A reutilização arriscada do mesmo contrato em vários EPGs sem compreender as regras de zoneamento resultantes pode rapidamente propagar-se em vários fluxos permitidos inesperadamente. Ao mesmo tempo, esses fluxos não intencionais continuarão a consumir a política CAM. Quando o CAM de política ficar cheio, a programação da regra de zoneamento começará a falhar, o que pode resultar em perda inesperada e intermitente, dependendo da configuração e dos comportamentos do endpoint.

Vazamento de VRF, tags de pc globais e direcionalidade de aplicação de política de L3Outs compartilhados

Este é um aviso especial para o caso de uso de serviços compartilhados que exige a configuração de contratos. Os serviços compartilhados normalmente implicam tráfego entre VRF em uma estrutura da ACI que depende do uso de um contrato com escopo 'locatário' ou 'global'. Para entender completamente isso, é preciso primeiro reforçar a ideia de que o valor típico de pcTag atribuído aos EPGs não é globalmente exclusivo. Os pcTags têm escopo definido para um VRF e o mesmo pcTag poderia ser reutilizado em outro VRF. Quando surgir a discussão sobre vazamento de rota, comece a aplicar os requisitos na estrutura da ACI, incluindo a necessidade de valores globalmente exclusivos, incluindo sub-redes e pcTags.

O que faz disso uma consideração especial é o aspecto de direcionalidade associado a um EPG ser um consumidor versus um provedor. Em um cenário de serviços compartilhados, normalmente se espera que o provedor impulsione um pcTag global para obter um valor exclusivo de malha. Ao mesmo tempo, o consumidor manterá seu pcTag com escopo de VRF, o que o coloca em uma posição especial para poder agora programar e entender o uso do valor global do pcTag para aplicar a política.

Como referência, o intervalo de alocação de pcTag é o seguinte:

- Sistema reservado: 1-15.
- Com escopo global: 16-16384 para EPGs de provedores de serviços compartilhados.
- Com escopo local: 16385-65535 para EPGs com escopo VRF.

Direção de aplicação de controle de política VRF

Em cada VRF é possível definir a configuração de direção de aplicação.

- A configuração padrão da direção de imposição é Ingress.
- A outra opção para direção de aplicação é Egress.

Entender onde a política é aplicada depende de várias variáveis diferentes.

A tabela abaixo ajuda a entender onde a política de segurança é aplicada no nível de folha.

Onde a política é aplicada?

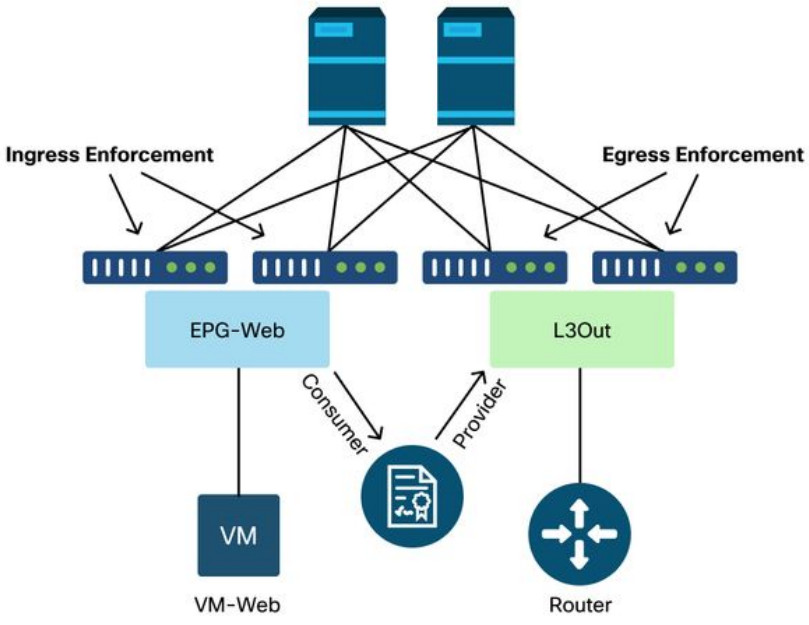
Cenário	modo de aplicação de VRF	Consumidor	Provedor	Política aplicada em
IntraVRF	Entrada/saída	EPG	EPG	·Se o endpoint de destino for aprendido: folha de entrada* ·Se o endpoint de destino não for

	Ingresso	EPG	EPG de saída L3	aprendido: folha de saída Folha de consumo (folha não fronteiriça)
	Ingresso	EPG de saída L3	EPG	Folha do provedor (folha fora da borda)
	Saída	EPG	EPG de saída L3	Borda leaf -> tráfego de folha não borda
	Saída	EPG de saída L3	EPG	·Se o endpoint de destino for aprendido: folha de borda ·Se o endpoint de destino não for aprendido: folha não borda Tráfego leaf-> border leaf ·Borda
	Entrada/saída	EPG de saída L3	EPG de saída L3	Folha de entrada*
	Entrada/saída	EPG	EPG	Folha de consumidor
	Entrada/saída	EPG	EPG de saída L3	Folha de consumo (folha não fronteiriça)
Inter-VRF	Entrada/saída	EPG de saída L3	EPG	Folha de entrada*
	Entrada/saída	EPG de saída L3	EPG de saída L3	Folha de entrada*

*A aplicação da política é aplicada na primeira folha atingida pelo pacote.

A figura abaixo ilustra um exemplo de aplicação de contrato em que o EPG-Web como consumidor e o L3Out EPG como provedor têm um contrato intraVRF. Se o VRF estiver definido no modo de imposição de entrada, a política será aplicada pelos nós de folha onde o EPG-Web reside. Se o VRF estiver definido no modo de imposição de saída, a política será aplicada pelos nós de folha de borda onde L3Out reside se o ponto de extremidade da VM-Web for aprendido na folha de borda.

Aplicação de entrada e saída



Ferramentas

Há uma variedade de ferramentas e comandos que podem ser usados para ajudar na identificação de uma **queda de política**. Uma queda de política pode ser definida como uma queda de pacote devido a uma configuração de contrato ou à falta dela.

Validação de regra de zoneamento

As ferramentas e comandos a seguir podem ser usados para validar explicitamente as regras de zoneamento que são programadas em switches leaf como resultado de relacionamentos de consumidor/provedor de contrato concluídos.

'show zoning-rules'

Um comando de nível de switch mostrando todas as regras de zoneamento em vigor.

```
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'

Um filtro que contém as informações de esporte/dport que a regra de zoneamento está executando. A programação de filtros pode ser verificada com este comando.

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

'show system internal policy-mgr stats'

Esse comando pode ser executado para verificar o número de acertos por regra de zoneamento. Isso é útil para determinar se uma regra esperada está sendo atingida em vez de outra, como uma regra de queda implícita que pode ter uma prioridade mais alta.

```
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'

Um comando no nível do switch que pode ser executado no nível do iBash que relata descartes relacionados à ACL (contrato) e informações relacionadas ao fluxo, incluindo:

- VRF
- VLAN-ID
- MAC origem/MAC destino
- IP de origem/IP de destino
- Porta origem/Porta destino
- Interface de origem

```
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

Um script Python no dispositivo que produz uma saída que correlaciona as regras de zoneamento, os filtros e as estatísticas de ocorrências ao executar pesquisas de nome a partir de IDs. Esse script é extremamente útil porque usa um processo de várias etapas e o transforma em um único comando que pode ser filtrado para EPGs/VRFs específicos ou em outros valores relacionados ao contrato.

```
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]
```

Validação de classificação de pacote

ELAM

Um relatório de nível ASIC usado para verificar detalhes de encaminhamento que indicam, no caso de um pacote descartado, o motivo da queda. Relevante para esta seção, o motivo pode ser um SECURITY_GROUP_DENY (queda da política do contrato).

Triagem

Um utilitário baseado em Python no APIC que pode rastrear o fluxo de pacotes de ponta a ponta com o ELAM.

Aplicativo Assistente do ELAM

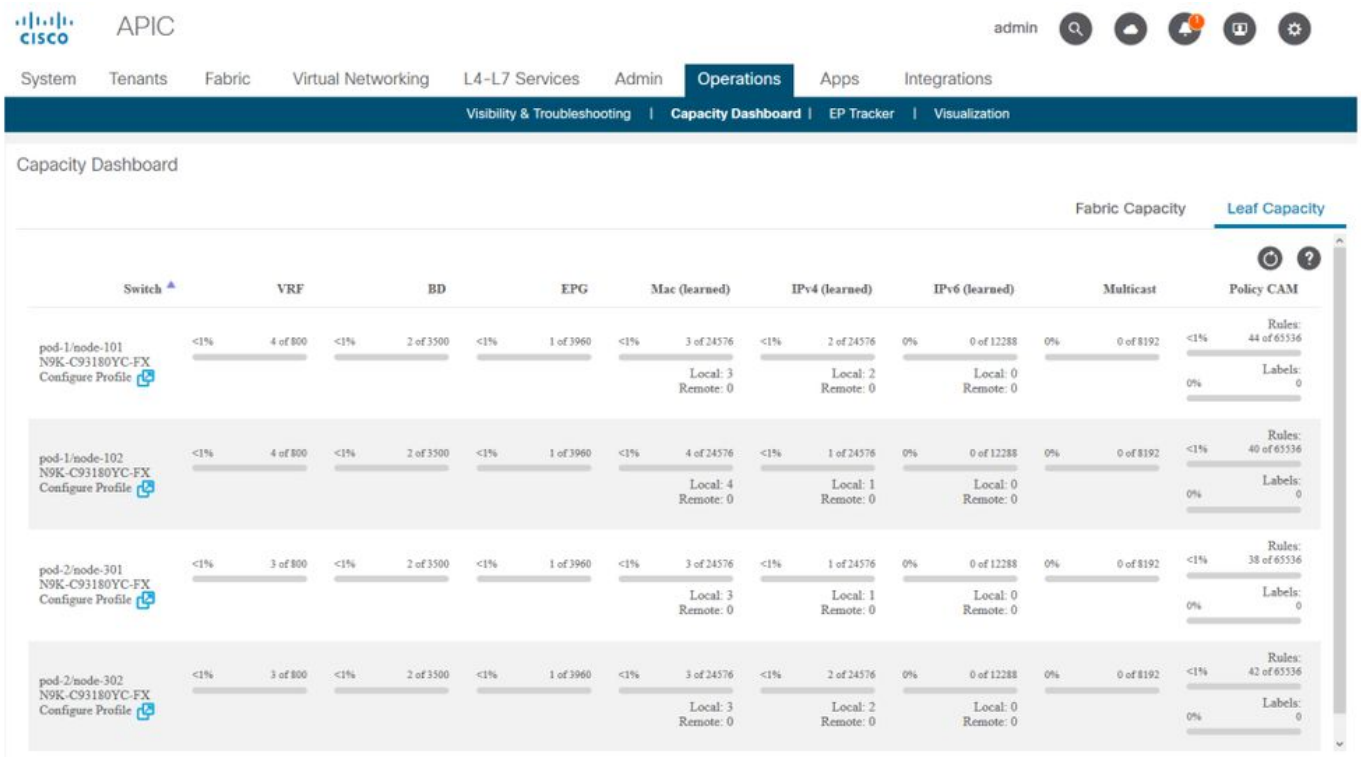
Um aplicativo APIC que abstrai a complexidade de vários ASICs para tornar a inspeção de decisão de encaminhamento muito mais conveniente e fácil de usar.

Consulte a seção "Encaminhamento de estrutura interna" para obter detalhes adicionais sobre as ferramentas ELAM, Triagem e Assistente ELAM

Uso de CAM de política

O uso da política CAM por folha é um parâmetro importante a ser monitorado para garantir que a estrutura esteja em um status íntegro. A maneira mais rápida de monitorar isso é usar o "painel de controle de capacidade" na GUI e verificar explicitamente a coluna "Policy Cam".

A exibição 'Capacidade Folha' do Painel de Controle de Capacidade



'show platform internal hal health-stats'

Este comando é útil para validar uma variedade de limites e uso de recursos, incluindo CAM de política. Observe que esse comando só pode ser executado em vsh_lc, portanto, transmita-o usando o sinalizador '-c' se estiver sendo executado do iBash.

```
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 para EPG

Considerações de descarte de política genérica

Há várias maneiras de solucionar um problema de conectividade entre dois endpoints. A metodologia a seguir fornece um bom ponto de partida para isolar rápida e efetivamente se o problema de conectividade é o resultado de uma **queda de política** (induzida por contrato).

Algumas perguntas de alto nível que vale a pena fazer antes de mergulhar em:

- Os endpoints estão no mesmo EPG ou em EPG diferentes? O tráfego entre dois endpoints que residem em EPGs diferentes (inter-EPG) é implicitamente negado e requer um contato para permitir a comunicação.O tráfego entre dois endpoints dentro do mesmo EPG (intra-EPG) é implicitamente permitido, a menos que o isolamento intra-EPG esteja em uso.
- O VRF é aplicado ou não? Quando um VRF está no modo **reforçado**, — dentro do VRF — os contratos são necessários para que os endpoints em dois EPGs diferentes se comuniquem.Quando um VRF está no modo **não aplicado**, — dentro do VRF — todo o tráfego seria permitido pela estrutura da ACI em vários EPGs pertencentes ao VRF não aplicado, independentemente dos contratos da ACI aplicados.

Metodologia

Com as várias ferramentas disponíveis, algumas são mais apropriadas e convenientes para começar do que outras, dependendo do nível de informação já conhecido sobre o fluxo afetado.

O caminho completo do pacote na estrutura da ACI é conhecido (folha de entrada, folha de saída...)?

- Se a resposta for sim, o Assistente ELAM deverá ser usado para identificar o motivo da queda no switch de origem ou destino.
- Se a resposta for não, Visibilidade e solução de problemas, Triagem, contract_parser, guia Operacional na exibição Locatário e comandos Bash ajudarão a restringir o caminho do pacote ou darão mais visibilidade aos motivos de queda.

Observe que a ferramenta Triagem não será discutida em detalhes nesta seção. Consulte o capítulo "Encaminhamento de estrutura interna" para obter mais detalhes sobre como usar essa ferramenta.

Considere que, embora a Visibilidade e Troubleshooting possa ajudar a visualizar rapidamente onde os pacotes são descartados entre dois pontos finais, a Triagem mostra informações mais detalhadas para Troubleshooting adicional. i.e. A triagem ajudará a identificar a interface, o motivo da queda e outros detalhes de baixo nível sobre o fluxo afetado

.

Este cenário de exemplo mostrará como solucionar problemas de queda de política entre dois endpoints: 192.168.21.11 e 192.168.23.11

Supondo que haja descartes de pacotes entre esses dois endpoints, o fluxo de trabalho de solução de problemas a seguir será usado para identificar a causa raiz do problema:

Identifique as folhas de src/dst envolvidas no fluxo de tráfego:

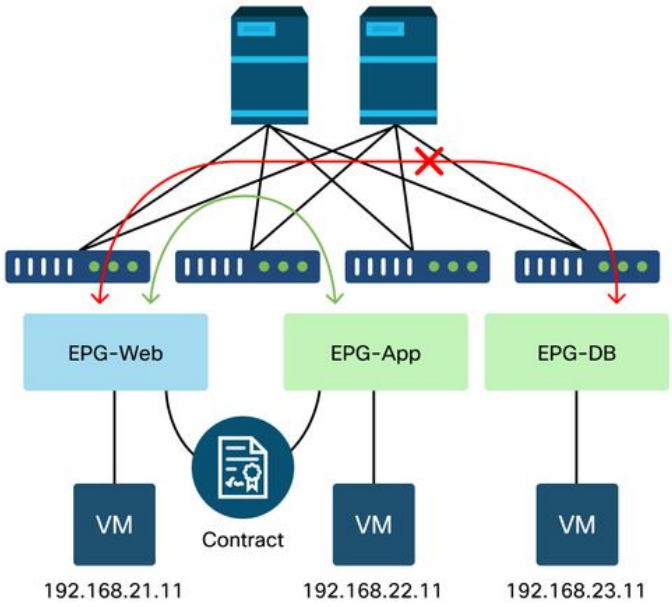
1. Use **Visibility & Troubleshooting** para rastrear o fluxo do pacote e identificar qual dispositivo está descartando o pacote.
2. Execute o comando 'show logging ip access-list internal packet-log deny' no dispositivo selecionado. Se um pacote com um dos endereços IP de interesse estiver sendo negado e registrado, o **registro de pacote** imprimirá o endpoint relevante e o nome do contrato por acerto.
3. Use o comando 'contract_parser.py —vrf <tenant>:<VRF>' no leaf de origem e de destino para observar a contagem de ocorrências do contrato configurado: Se um pacote estiver atingindo o contrato no switch de origem ou de destino, o contador do contrato relevante será incrementadoEsse método é menos granular que o do registro de pacote interno da lista de acesso IP em situações em que muitos fluxos podem atingir a mesma regra (muitos endpoints/fluxos entre os dois EPGs de interesse).

As etapas acima são descritas com mais detalhes no próximo parágrafo.

Exemplo de cenário de Troubleshooting de EPG para EPG

Este cenário de exemplo mostrará como solucionar problemas de queda de política entre dois endpoints: 192.168.21.11 no EPG-Web e 192.168.23.11 no EPG-DB.

Topologia

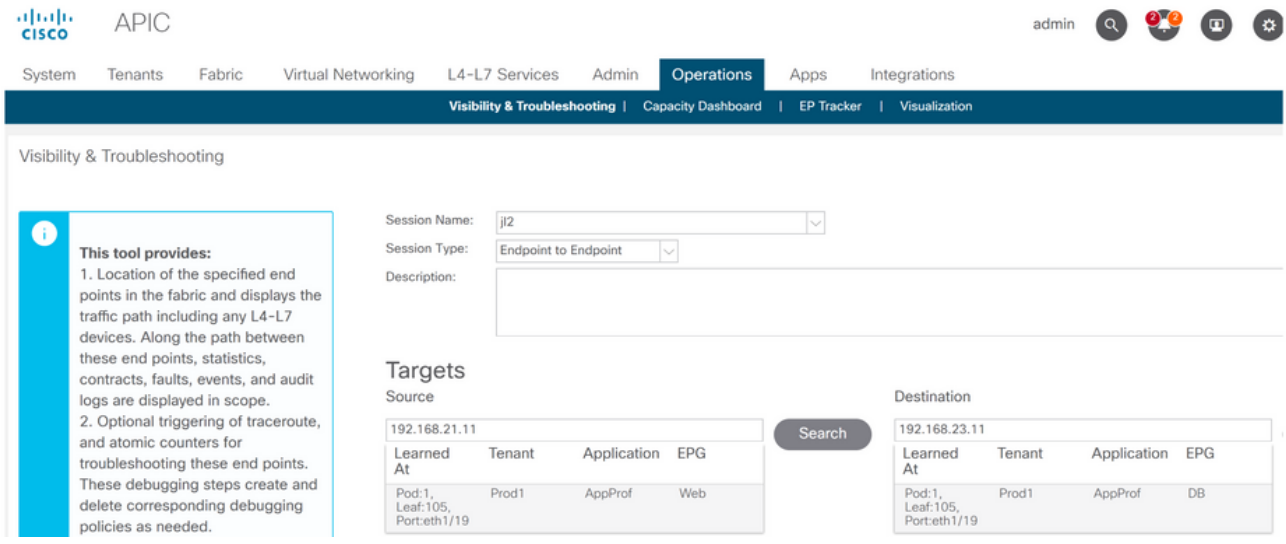


Identificar os switches leaf de origem e destino envolvidos no descarte de pacotes

Visibilidade e solução de problemas

A ferramenta Visibility & Troubleshooting (Visibilidade e solução de problemas) ajudará a visualizar o switch onde ocorreu a queda de pacote para um fluxo EP-para-EP específico e a identificar onde os pacotes possivelmente foram descartados.

Configuração de visibilidade e solução de problemas

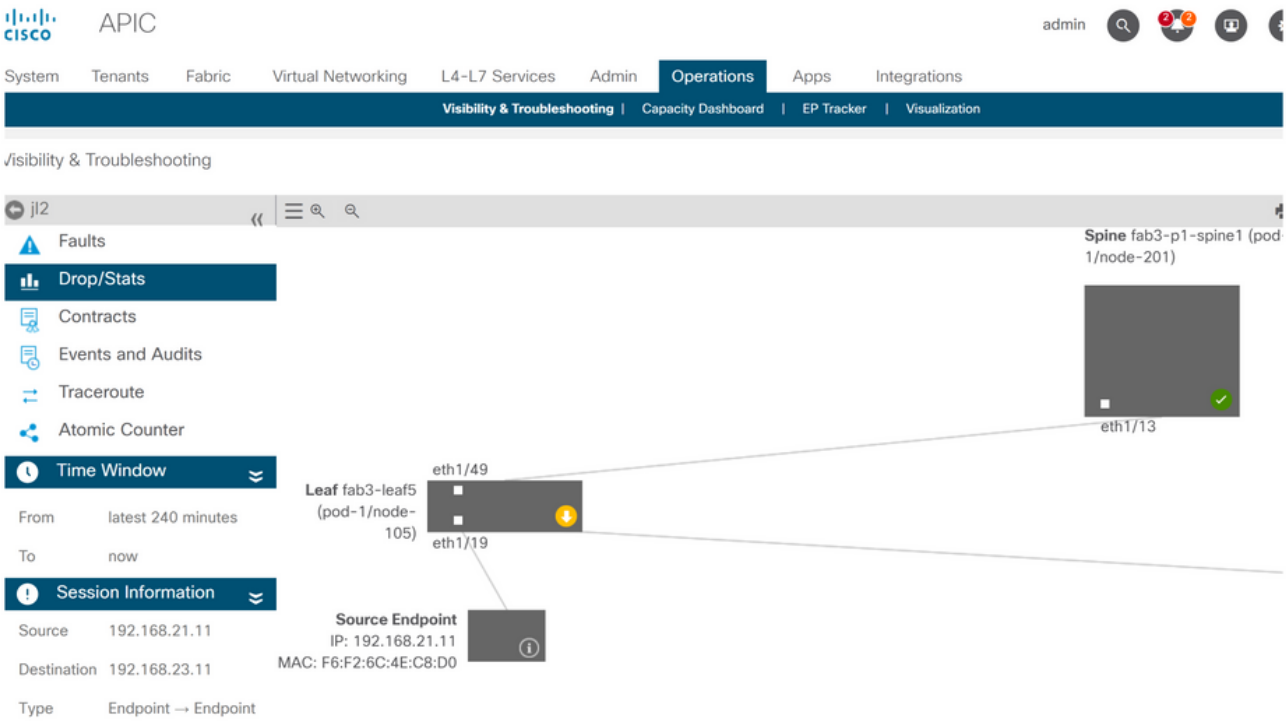


Configure um Nome de Sessão, uma Origem e um endpoint de Destino. Em seguida, clique em 'Enviar' ou em 'Gerar relatório'.

A ferramenta localizará automaticamente os endpoints na malha e fornecerá informações sobre o usuário, o perfil do aplicativo e o EPG aos quais o EP pertence.

Nesse caso, descobrirá que os EPs pertencem ao locatário Prod1, pertencem ao mesmo perfil de aplicativo 'AppProf' e são atribuídos a diferentes EPGs: 'Web' e 'DB'.

Identificação de queda

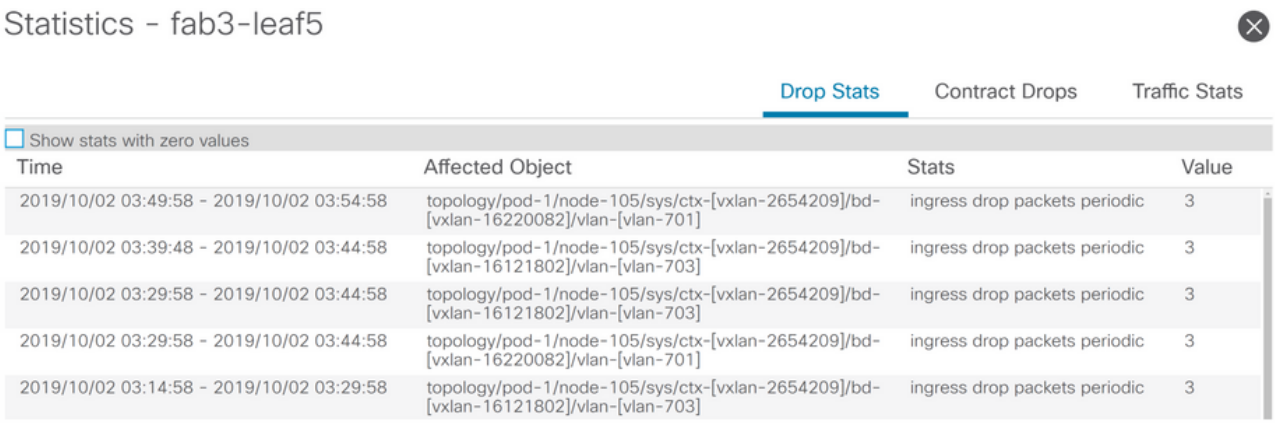


A ferramenta visualizará automaticamente a topologia do cenário de solução de problemas. Nesse caso, os dois pontos finais estão conectados ao mesmo switch leaf.

Navegando até o submenu Eliminar/Estatísticas, o usuário pode visualizar as eliminações gerais na folha ou na coluna em questão. Consulte a seção "Quedas de interface" no capítulo "Encaminhamento de estrutura interna" deste manual para obter mais informações sobre como entender quais quedas são relevantes.

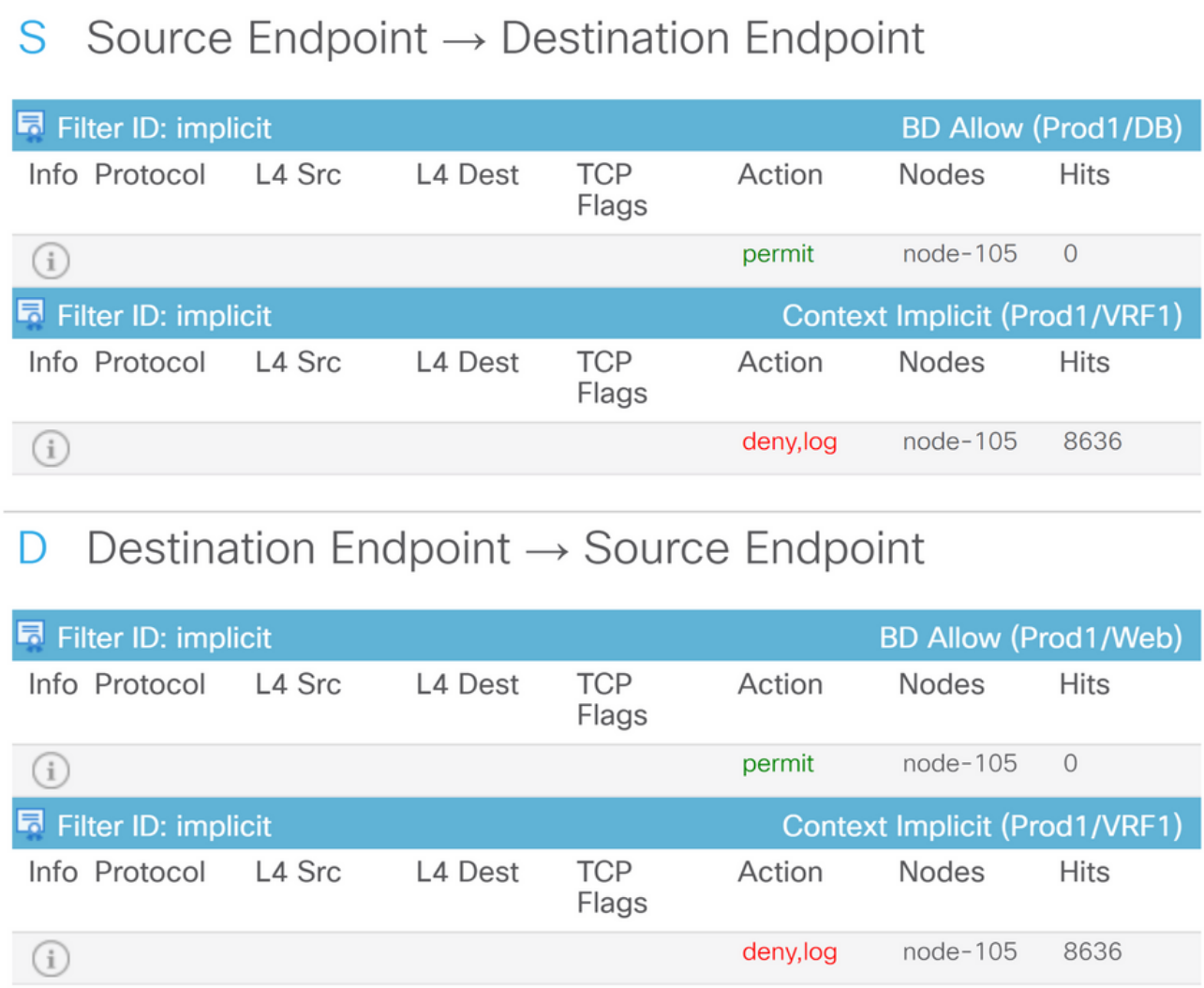
Muitas dessas quedas são comportamento esperado e podem ser ignoradas.

Remover detalhes



Ao fazer drill-down para soltar detalhes usando o botão amarelo 'Pacotes soltos' no diagrama do switch, o usuário pode exibir detalhes sobre o fluxo solto.

Detalhes do contrato



Navegando até o submenu Contratos, o usuário pode identificar qual contrato está causando a queda da política entre os EPGs. No exemplo, é Implícito negar Prod1/VRF1, que mostra alguns acertos. Isso não significa necessariamente que o fluxo especificado (192.168.21.11 e 192.168.23.11) está atingindo esse deny implícito. Se a regra Hits of Context Implicit deny estiver aumentando, isso significa que há tráfego entre Prod1/DB e Prod1/Web que não atinge nenhum dos contratos, portanto, é descartado pela instrução deny implícita.

Na exibição Topologia do perfil de aplicativo em Locatário > selecione o nome do perfil de aplicativo à esquerda > Topologia , é possível verificar quais contratos estão aplicados ao DB EPG. Nesse caso, nenhum contrato é atribuído ao EPG:

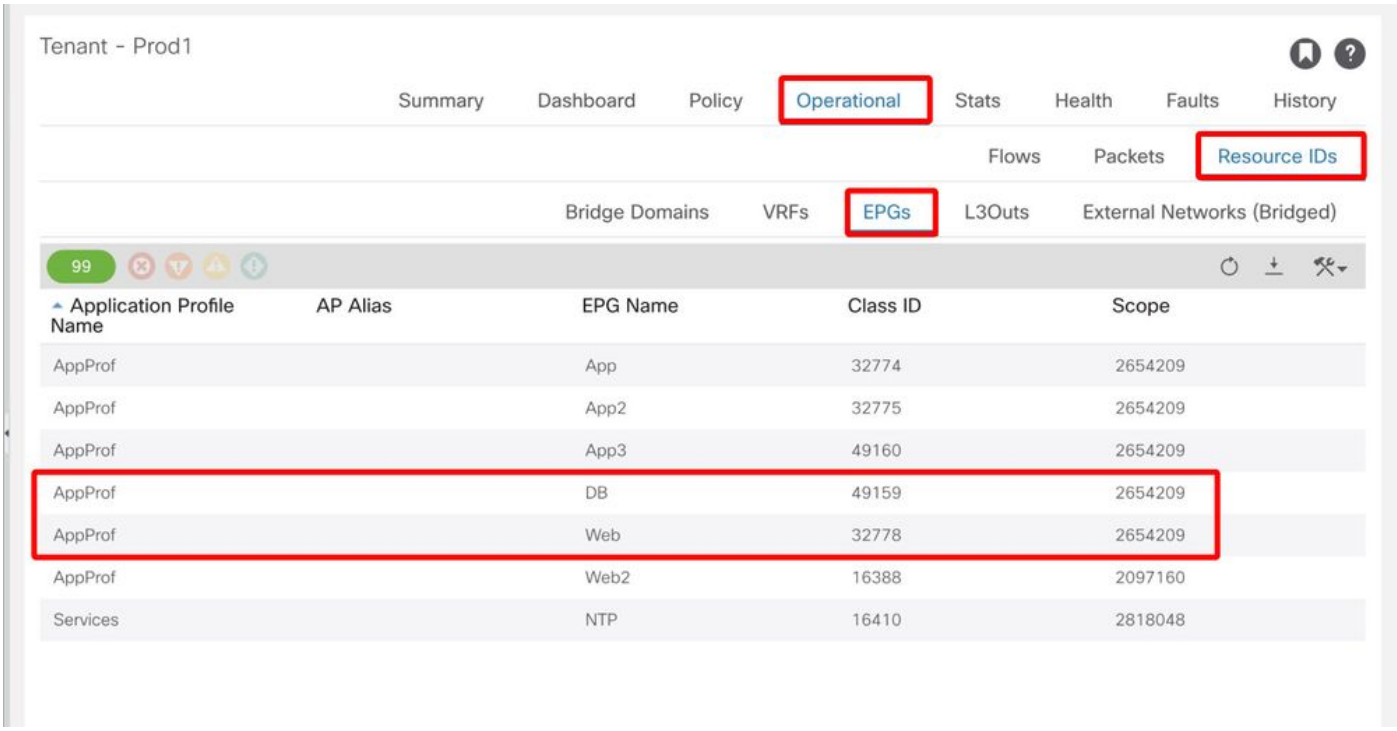
Visualização do contrato

Agora que os EPGs de origem e destino são conhecidos, também é possível identificar outras informações relevantes, como:

- O **pcTag** src/dst **EPG** dos endpoints afetados. O pcTag é a ID de classe usada para identificar um EPG com uma regra de zoneamento.
- O **VRFVNID** src/dst, também conhecido como **escopo**, dos endpoints afetados.

A ID de classe e o escopo podem ser facilmente recuperados da GUI do APIC abrindo o Espaço > selecione o nome do Espaço à esquerda > Operacional > IDs de recursos > EPGs

ID do recurso do locatário para localizar pcTag e escopo do EPG



Nesse caso, a ID da classe e os escopos são:

- 32778 do pcTag do Web EPG
- 2654209 de escopo de EPG da Web
- 49159 de pcTag de DB EPG
- 2654209 de escopo de EPG de BD

Verifique a política aplicada ao fluxo de tráfego que está sendo solucionado

iBash

Uma ferramenta interessante para verificar o pacote descartado em uma folha da ACI é a linha de comando Bash: '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
```

Conforme a saída anterior, pode-se ver que no switch leaf, vários pacotes ICMP originados pelo EP 192.168.23.11 em direção a 192.168.21.11 foram descartados.

A ferramenta contract_parser ajudará a verificar as políticas reais aplicadas ao VRF às quais os endpoints estão associados:

```
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]
```

Isso também pode ser verificado por meio da regra de zoneamento programada na folha e das políticas aplicadas pelo switch.

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

Como já visto pela ferramenta Visibility & Troubleshooting, pela ferramenta contract_parser e pelas regras de zoneamento, a saída confirma que não há contrato entre os EPGs de origem e de destino na solução de problemas. É fácil supor que os pacotes descartados correspondem à regra de negação implícita 5155.

Captura ELAM

A captura ELAM fornece um relatório de nível ASIC usado para verificar detalhes de encaminhamento que indicam, no caso de um pacote descartado, o motivo da queda. Quando o motivo de uma queda for uma queda de política, como neste cenário, a saída da captura ELAM será semelhante à seguinte.

Observe que os detalhes de configuração de uma captura ELAM não serão discutidos neste capítulo. Consulte o capítulo "Encaminhamento intradfabric".

```
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
```

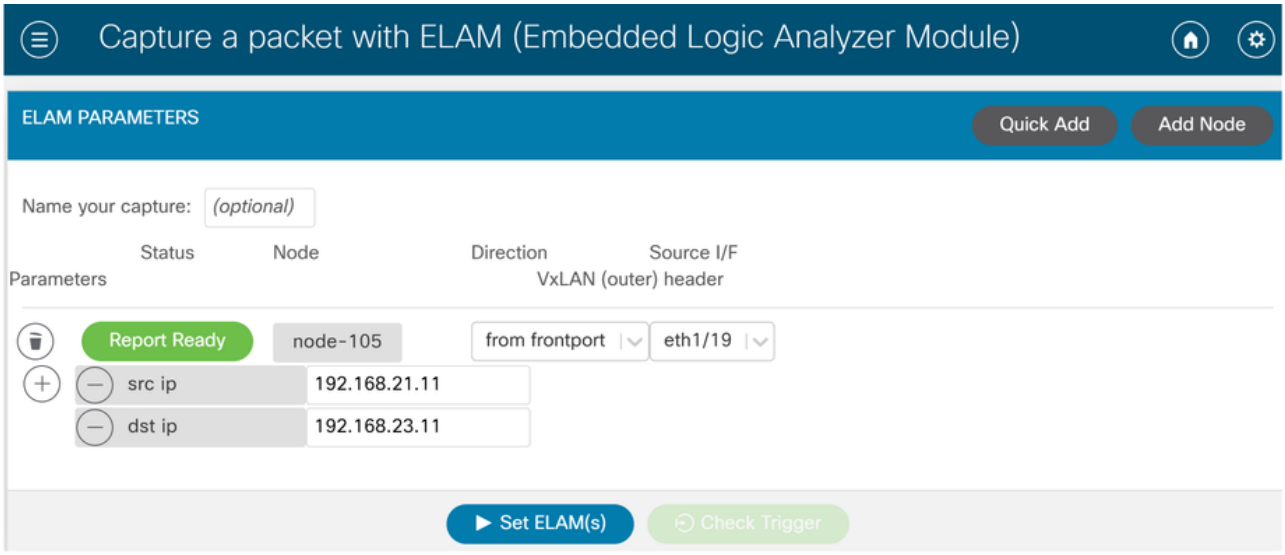
```
pkt.lu_drop_reason: 0x2D
```

O relatório ELAM acima mostra claramente que o pacote foi descartado devido a uma queda de política: 'SECURITY_GROUP_DENY'

Assistente do ELAM:

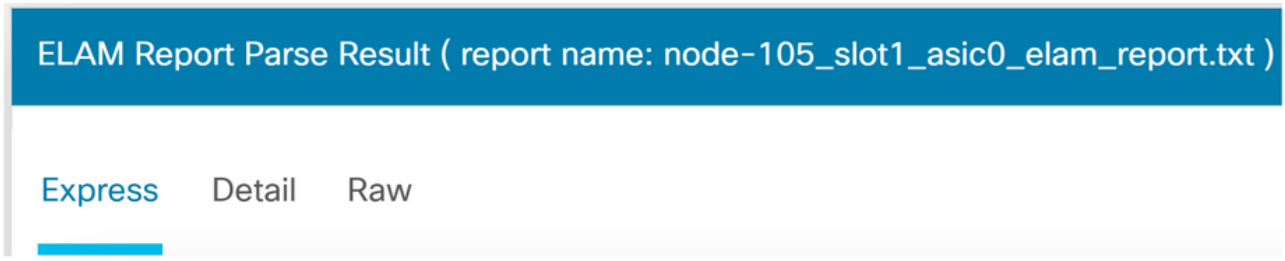
O mesmo resultado da captura do ELAM pode ser mostrado através do aplicativo ELAM Assistant na GUI do APIC.

Configuração



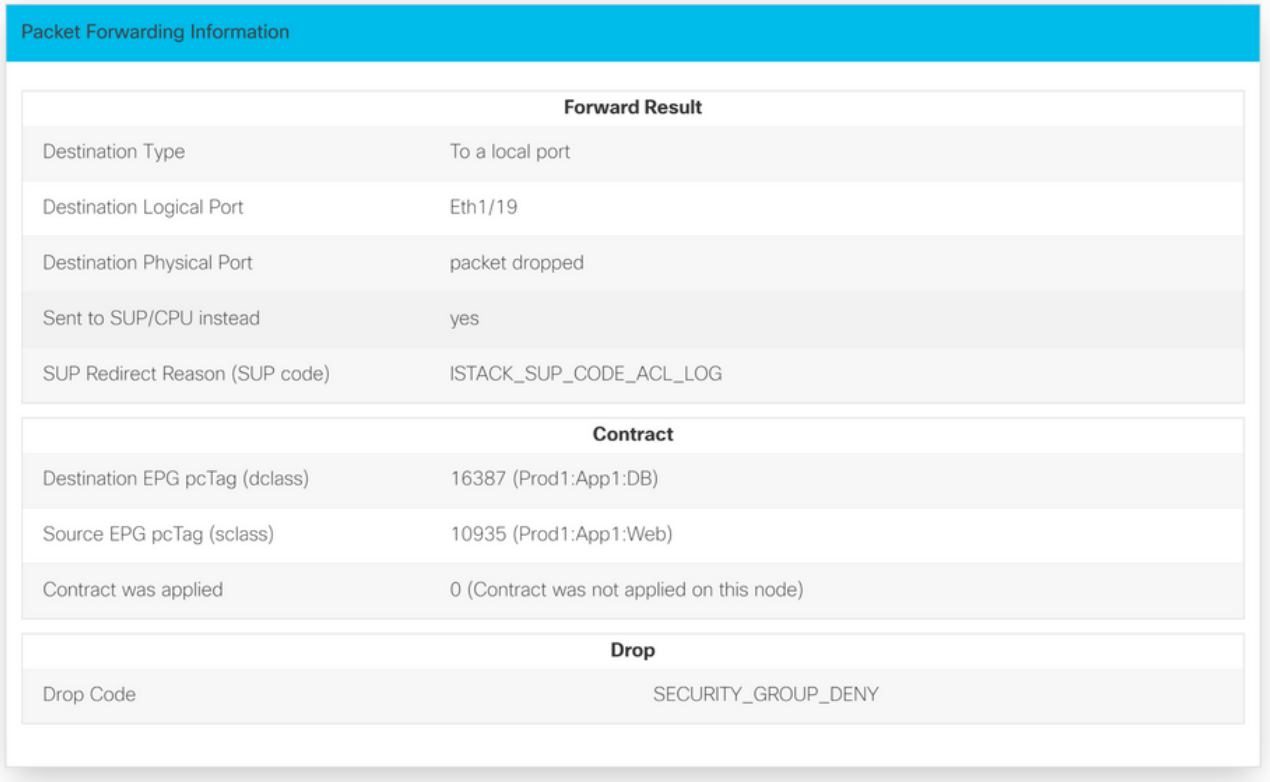
Normalmente, o usuário configurará os detalhes de origem e destino para o fluxo de interesse. Neste exemplo, o IP src é usado para capturar o tráfego para o ponto final no EPG de destino que não tem um relacionamento de contrato com o EPG de origem.

Relatório Elam Assistant Express



Há três níveis de saída que podem ser exibidos com o Assistente do ELAM. Eles são Express, Detail e Raw.

Relatório Elam Assistant Express (cont.)



Packet Forwarding Information

Forward Result	
Destination Type	To a local port
Destination Logical Port	Eth1/19
Destination Physical Port	packet dropped
Sent to SUP/CPU instead	yes
SUP Redirect Reason (SUP code)	ISTACK_SUP_CODE_ACL_LOG

Contract	
Destination EPG pcTag (dclass)	16387 (Prod1:App1:DB)
Source EPG pcTag (sclass)	10935 (Prod1:App1:Web)
Contract was applied	0 (Contract was not applied on this node)

Drop	
Drop Code	SECURITY_GROUP_DENY

Em Express Result (Resultado expresso), a razão do código de queda SECURITY_GROUP_DENY indica que a queda foi resultado de um acerto de contrato.

Grupo preferido

Sobre grupos preferenciais de contrato

Há dois tipos de aplicação de política disponíveis para EPGs em um VRF com um grupo de preferência de contrato configurado:

- EPGs incluídos: Os EPGs podem se comunicar livremente entre si sem contratos, se tiverem participação em um grupo de preferência de contrato. Isso é baseado na regra padrão source-any-destination-any-permit.
- EPGs excluídos: Os EPGs que não são membros de grupos preferenciais exigem que os contratos se comuniquem entre si. Caso contrário, as regras de negação entre o EPG excluído e qualquer EPG serão aplicadas.

O recurso de grupo preferido pelo contrato permite maior controle da comunicação entre EPGs em um VRF. Se a maioria dos EPGs no VRF deve ter comunicação aberta, mas alguns devem ter comunicação limitada apenas com os outros EPGs, configure uma combinação de um grupo preferido de contrato e contratos com filtros para controlar com mais precisão a comunicação entre EPGs.

Os EPGs excluídos do grupo preferencial só poderão se comunicar com outros EPGs se houver um contrato em vigor para substituir a regra padrão source-any-destination-any-deny.

Programação do grupo de preferência do contrato

Essencialmente, os Grupos de Contrato Preferenciais são um inverso dos contratos regulares. Para contratos regulares, as regras de zoneamento de permissão explícita são programadas com

uma regra de zoneamento deny implícita com o escopo VRF. Para grupos preferenciais, uma regra de zoneamento PERMIT implícita é programada com o valor de prioridade numérica mais alto e as regras de zoneamento DENY específicas são programadas para não permitir o tráfego de EPGs que não sejam membros do grupo preferencial. Como resultado, as regras de negação são avaliadas primeiro e, se o fluxo não corresponder a essas regras, o fluxo será implicitamente permitido.

Há sempre um par de regras de zoneamento deny explícito para cada EPG fora do grupo preferido:

- Um do membro do Grupo não Preferencial para qualquer pcTag (valor 0).
- Outro de qualquer pcTag (valor 0) para o membro do Grupo não Preferencial.

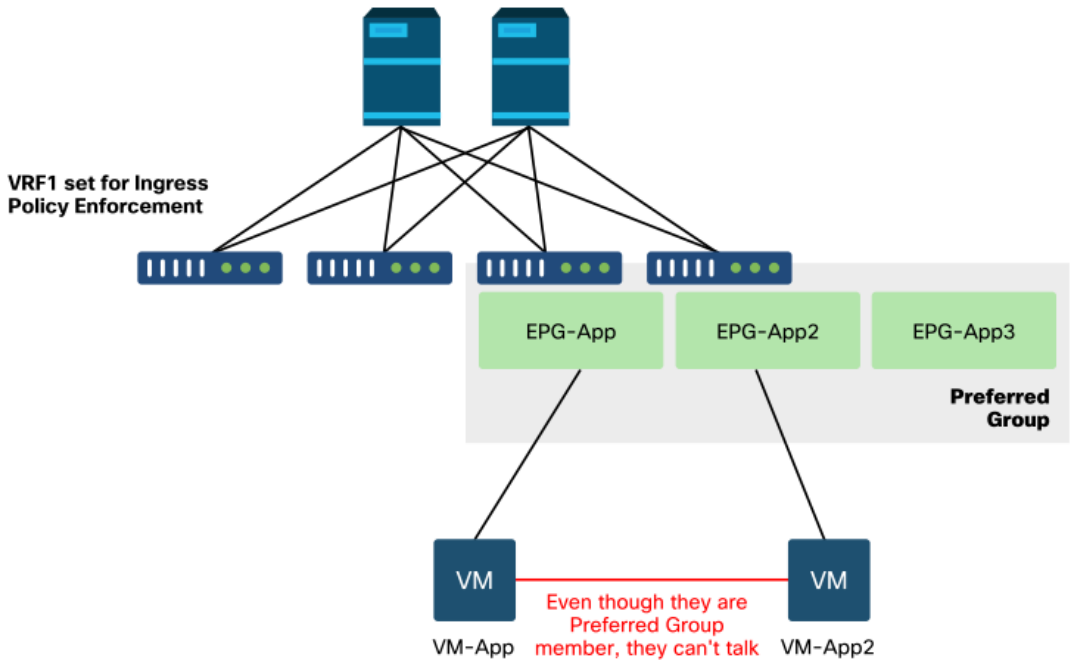
Cenário de Troubleshooting do Grupo Preferido

A figura abaixo mostra uma topologia lógica na qual os EPGs App, App2 e App3 estão configurados como Preferred Group Members.

O VM-App faz parte do EPG-App e o VM-App2 faz parte do EPG-App2. O EPG do App e do App2 deve fazer parte do EPG preferencial e, portanto, comunicar-se livremente.

O VM-App inicia um fluxo de tráfego na porta TCP 6000 para o VM-App2. O EPG-App e o EPG-App2 são membros do grupo preferencial como parte do VRF1. O VM-App2 nunca recebe pacotes na porta TCP 6000.

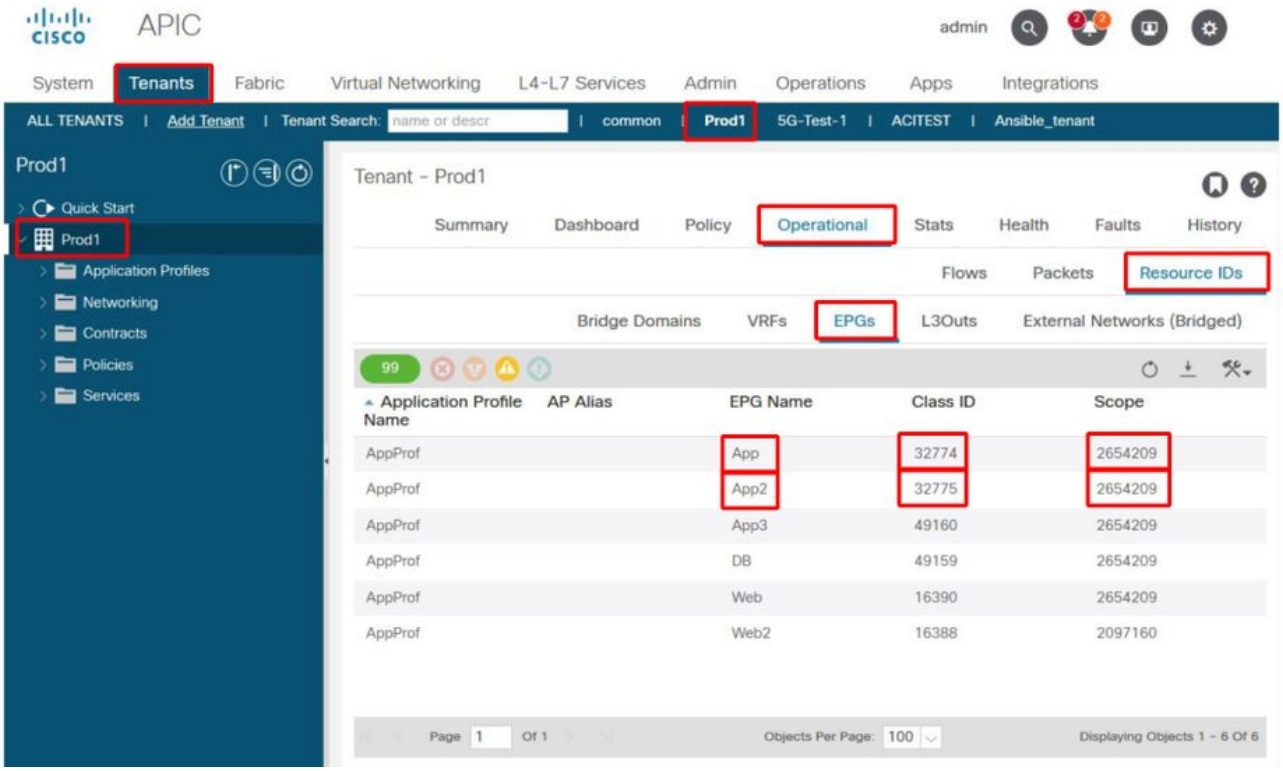
Topologia



Fluxo de trabalho

1. Procure o pcTag do EPG APP e seu VNID/Escopo VRF

EPG e VRF pcTags



2. Verifique a programação do contrato usando contract_parser.py na folha de entrada

Use contract_parser.py e/ou o comando 'show zoning-rule' e especifique o 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]
```

Examinando a saída acima, a entrada permit implícita — ruleId 4165 — com a prioridade mais alta de 20 é observada. Essa regra de permissão implícita fará com que todos os fluxos de tráfego sejam permitidos, a menos que haja uma regra de negação explícita com uma prioridade mais baixa impedindo o fluxo de tráfego.

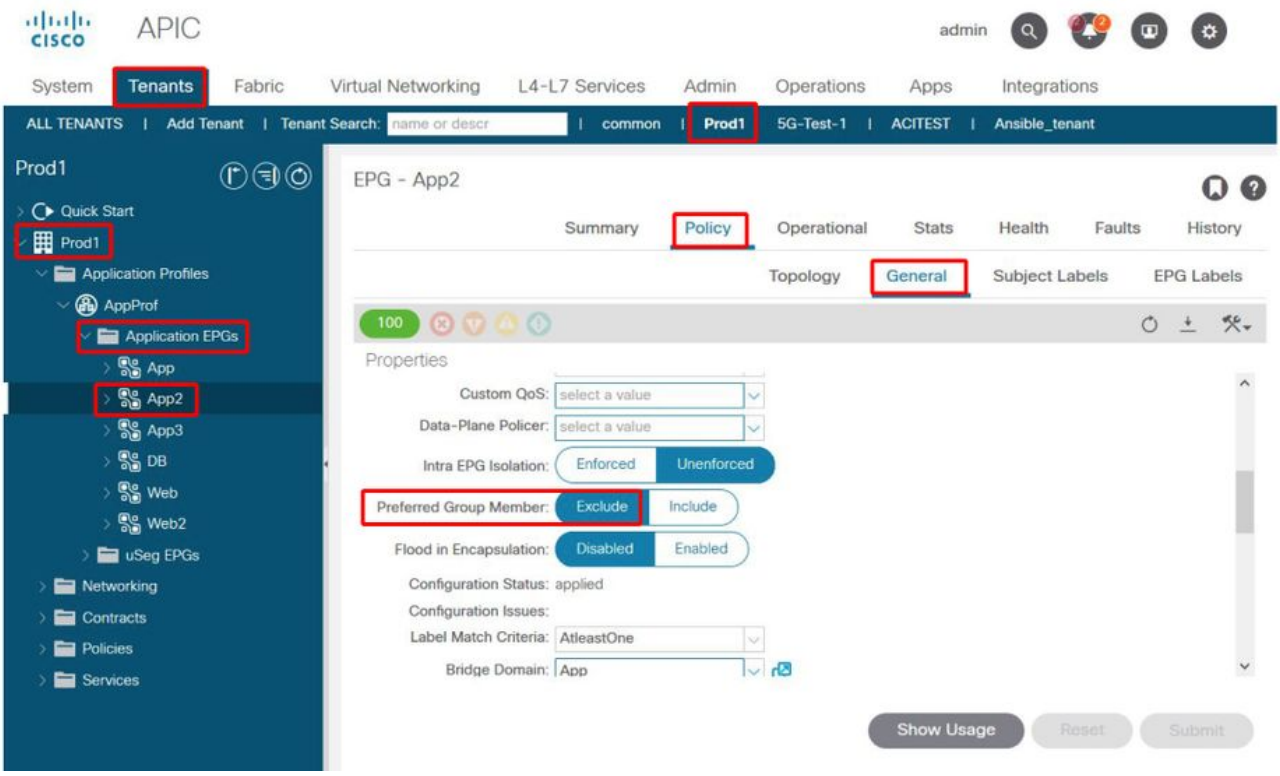
Além disso, há duas regras explícitas de negação observadas para o pcTag 32775, que é o pcTag do EPG App2. Essas duas regras explícitas de negação de zoneamento não permitem o tráfego de qualquer EPG para o EPG App2 e vice-versa. Essas regras têm prioridade 18 e 19, portanto elas terão prioridade sobre a regra de permissão padrão.

A conclusão é que o EPG App2 não é um membro do grupo preferido, pois as regras explícitas de negação são observadas.

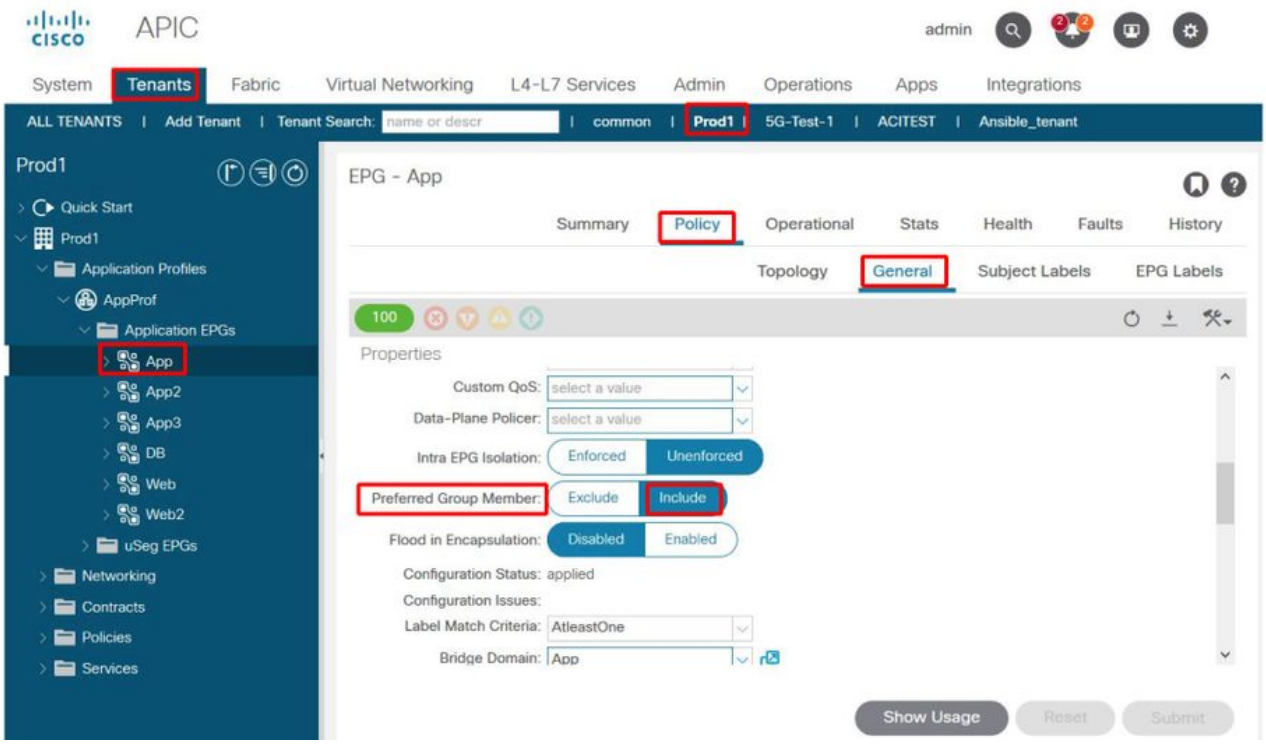
3. Verificar a Configuração do Membro do Grupo preferido no EPG

Navegue pela GUI do APIC e verifique EPG App2 e EPG App Preferred Group Member Configuration. Na figura a seguir, consulte EPG App2 não está configurado como um Preferred Group Member.

EPG App2 — Configuração de Membro de Grupo Preferido excluída



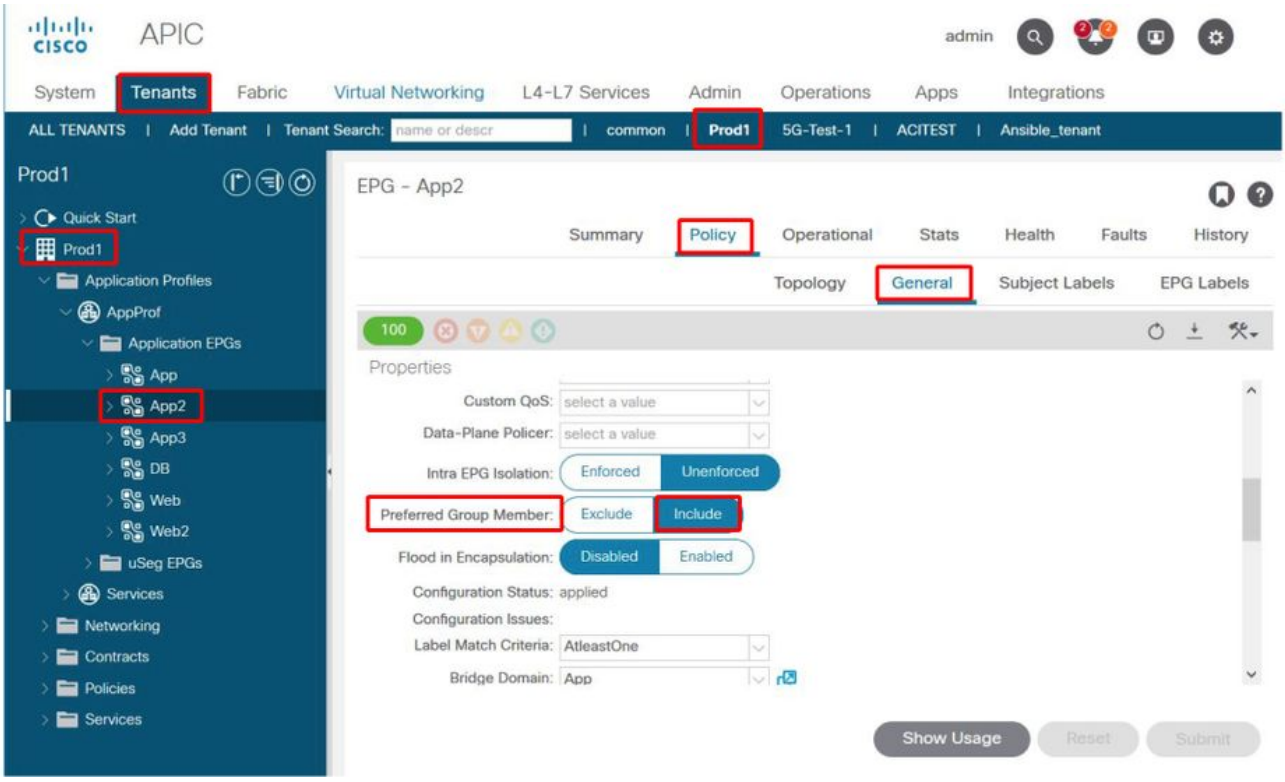
Aplicativo EPG — Configuração de membro de grupo preferencial incluída



4. Definir o EPG App2 como um Membro do Grupo Preferido

Alterar a configuração do App2 EPG permite que o grupo preferencial se comunique livremente como parte do grupo preferencial.

EPG App2 — Configuração de Membro de Grupo Preferido incluída



5. Verifique novamente a programação do contrato usando contract_parser.py na folha onde o EP src reside

Use contract_parser.py novamente e especifique o nome do VRF para verificar se as regras de negação explícitas para o EPG App2 foram removidas.

```
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]
```

As regras explícitas de negação para EPG App2 e seus 32775 pcTag não são mais observadas na saída acima. Isso significa que o tráfego entre EPs no EPG App e no EPG App2 agora corresponderá à regra de permissão implícita — ruleId 4165 — com a prioridade mais alta de 20.

vzAny para EPG

Sobre vzAny

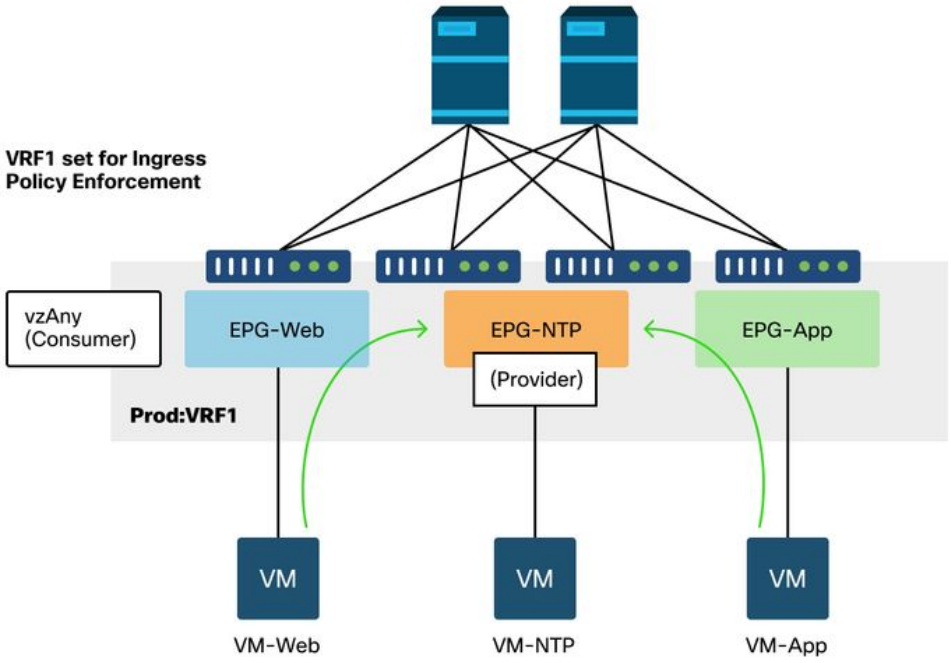
Ao configurar contratos entre um ou vários EPGs, os contratos podem ser configurados como uma relação consumida ou fornecida. Quando o número de EPGs aumenta, o mesmo ocorre com

a quantidade de relações contratuais entre eles. Alguns casos de uso comuns exigem que todos os EPGs troquem fluxos de tráfego com outro EPG específico. Esse caso de uso pode ser um EPG contendo EPs que fornecem serviços que precisam ser consumidos por todos os outros EPGs dentro do mesmo VRF (NTP ou DNS, por exemplo). O vzAny permite uma menor sobrecarga operacional na configuração de relações contratuais entre todos os EPGs e EPGs específicos que fornecem serviços a serem consumidos por todos os outros EPGs. Além disso, o vzAny permite um uso muito mais eficiente do CAM de política de segurança em switches leaf, pois apenas 2 regras de zoneamento são adicionadas para cada relação de contrato do vzAny.

Exemplo de caso de uso

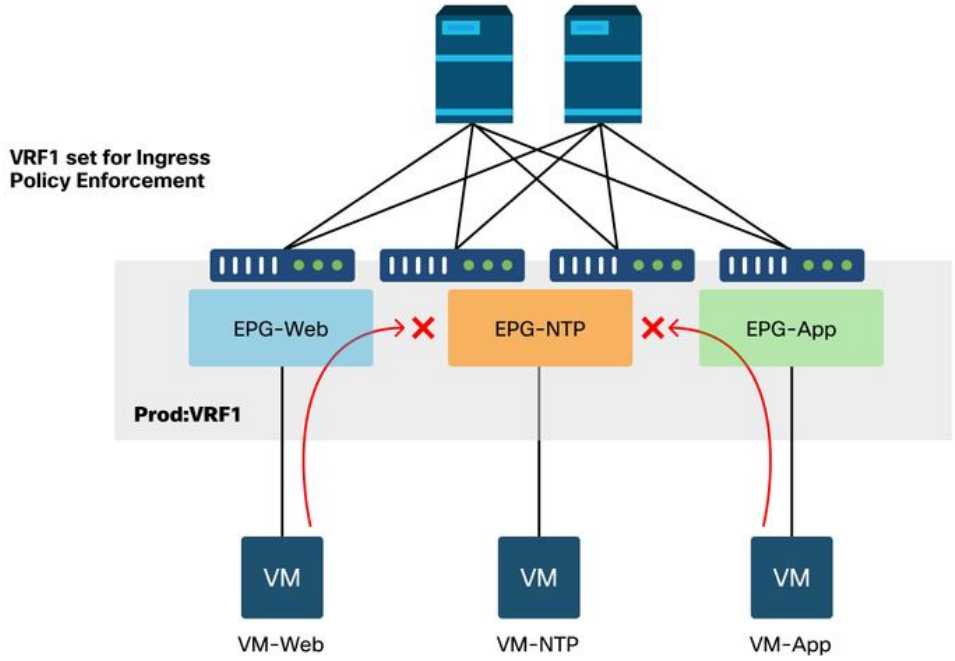
A figura abaixo descreve esse caso de uso em que VM-Web e VM-App em EPGs Web e App respectivamente precisam consumir serviços NTP de VM-NTP em EPG-NTP. Em vez de configurar um contrato fornecido no EPG NTP e, subsequentemente, ter esse mesmo contrato como um contrato consumido no EPGs Web e App, o vzAny permite que cada EPG no VRF Prod:VRF1 consuma serviços NTP do EPG NTP.

vzAny — Qualquer EPG no VRF Prod:VRF1 pode consumir serviços NTP do EPG NTP



Considere um cenário onde quedas são observadas entre EPGs que consomem os serviços NTP quando não há contrato entre eles.

Cenário de Troubleshooting - O tráfego cai se não houver contrato

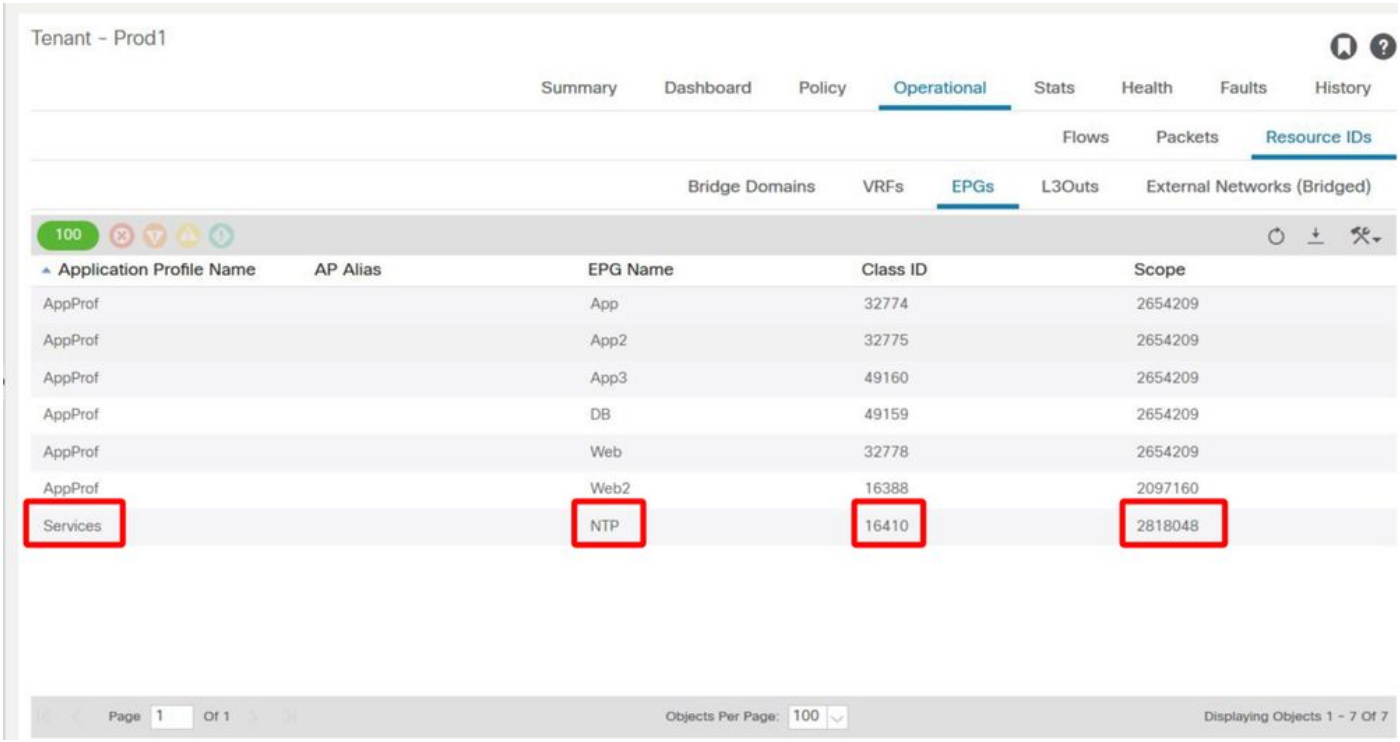


Fluxo de trabalho

1. Pesquise o pcTag do EPG NTP e seu VNID/Escopo VRF

'Locatário > Operacional > IDs de recursos > EPGs' permite localizar o pcTag e o escopo

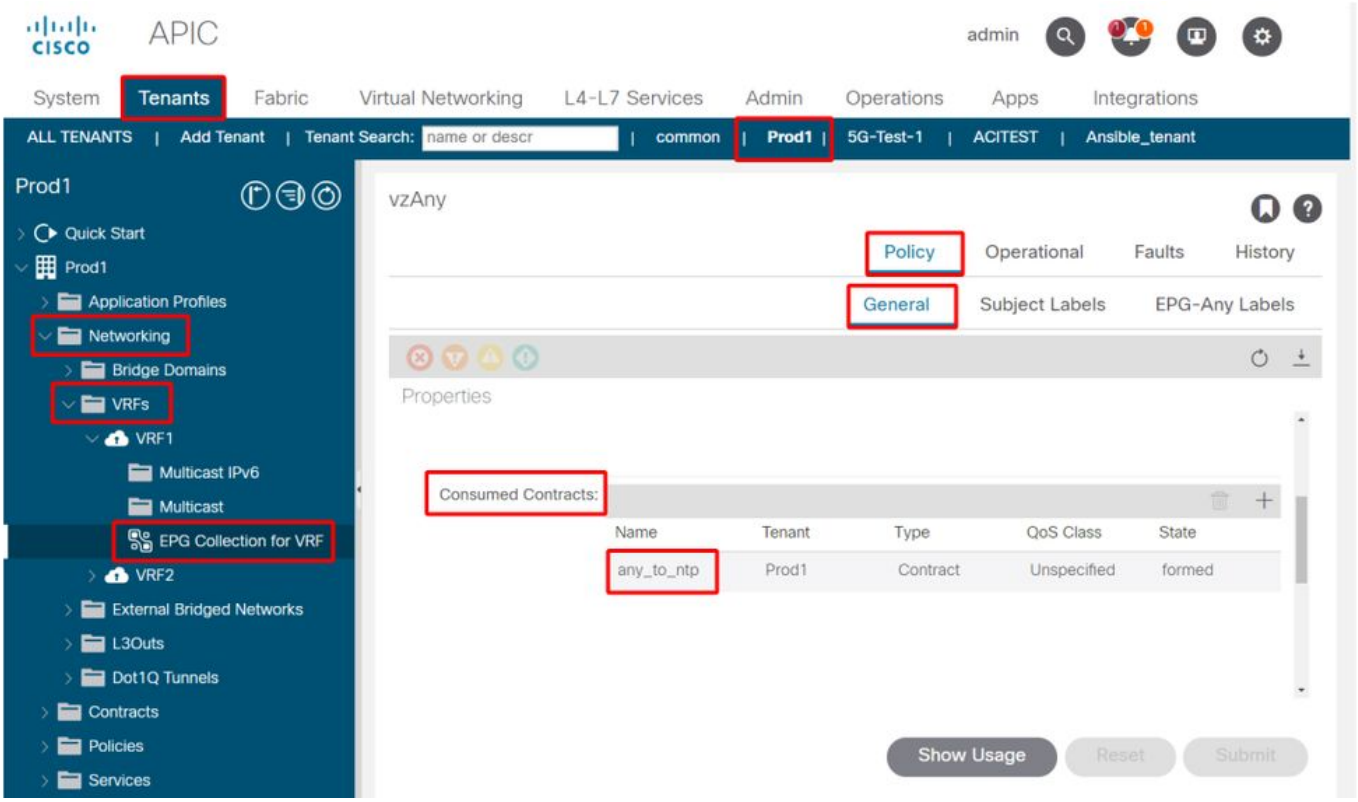
EPG NTP pcTag e seu VRF VNID/Scope



2. Verificar se um contrato está configurado como vzQualquer contrato consumido como parte do VRF

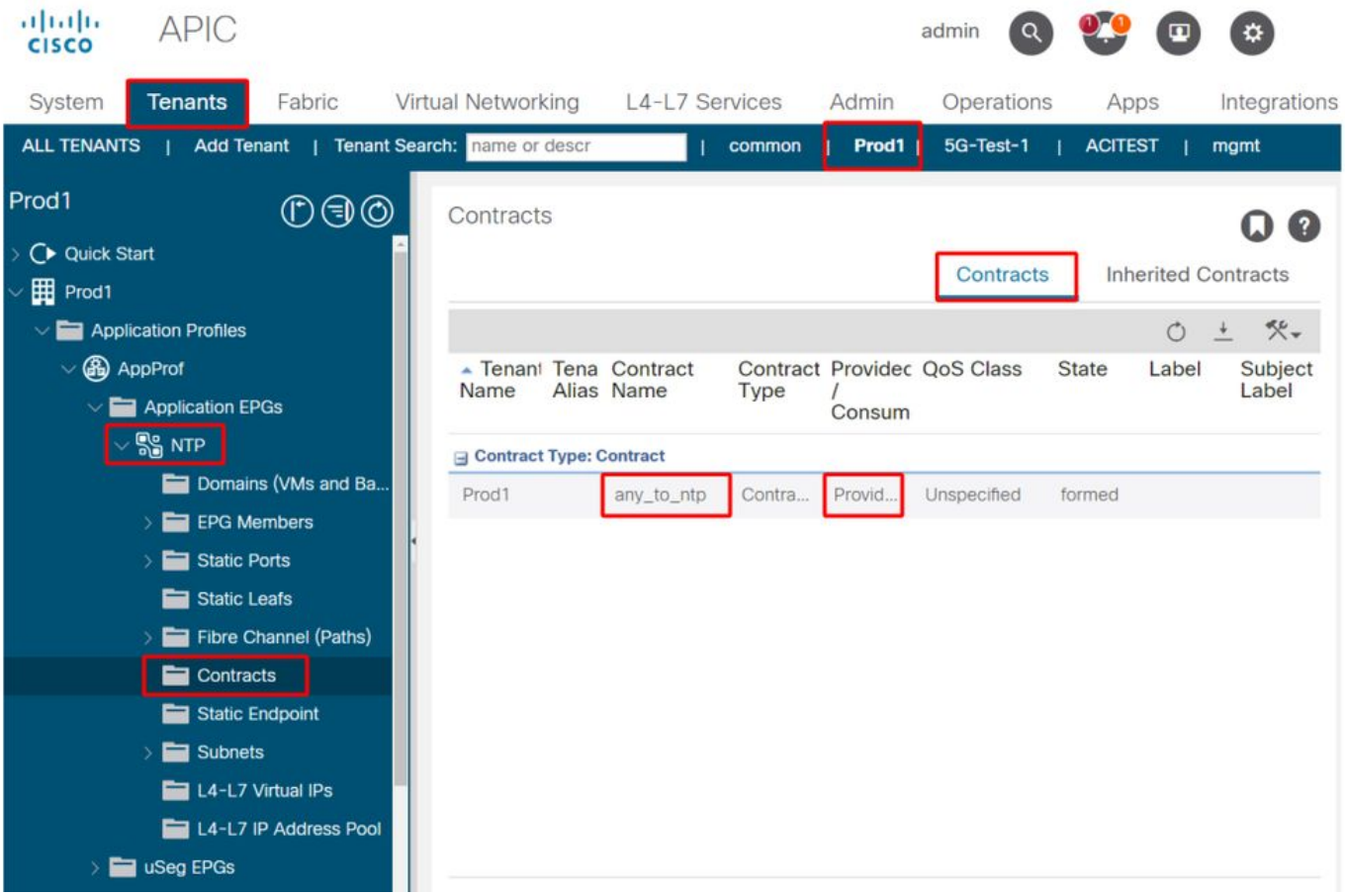
Navegue até o VRF e verifique se há um contrato consumido configurado como vzAny em 'EPG Collection for VRF'.

Contrato configurado como um contrato vzAny consumido no VRF



3. Verifique se o mesmo contrato é aplicado como um contrato fornecido no EPG NTP

Para estabelecer uma relação contratual, o mesmo contrato deve ser aplicado como um contrato fornecido no EPG NTP, que fornece serviços NTP aos outros EPGs em seu VRF.



4. Verificação de regra de zoneamento na folha de ingresso usando contract_parser.py ou 'show zoning-rule'

A folha de entrada deve ter 2 regras de zoneamento para permitir fluxos de tráfego bidirecionais (se o objeto do contrato estiver definido para permitir ambas as direções) entre qualquer EPG e EPG NTP. 'Qualquer EPG' é indicado como pcTag 0 na programação de regra de zoneamento.

O uso do contract_parser.py ou dos comandos 'show zoning-rule' na folha de entrada durante a especificação do VRF permite garantir que a regra de zoneamento seja programada.

Regras de zoneamento que permitem o tráfego de/para o EPG NTP de outros EPGs no VRF presente

Usando contract_parser.py e 'show zoning-rule' para verificar a presença de vzAny based zoning-rules.

Dois tipos de regras são evidentes:

1. As regras 4156 e 4168, que permitem que qualquer pessoa se torne NTP e vice-versa. Têm prioridade 13 e 14: Regra de zoneamento que permite fluxos de tráfego de qualquer EPG (pcTag 0) para o EPG NTP (pcTag 49161).Regra de zoneamento que permite fluxos de tráfego de EPG NTP (pcTag 46161) para qualquer outro EPG (pcTag 0).
2. Regra 4165, que é a regra deny any to any (padrão) com prioridade 21.

Como a prioridade mais baixa tem precedência, todos os EPGs do VRF terão acesso ao EPG do NTP.

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

L3Out Compartilhado para EPG

Sobre L3Out Compartilhado

Saída de Camada 3 Compartilhada é uma configuração que permite ter uma Saída L3em um VRF fornecendo alguns serviços (acesso externo) e um ou mais outros VRFs consumirem essa Saída L3D. Mais detalhes sobre L3Out compartilhado podem ser encontrados no capítulo "Roteamento externo".

Ao fazer L3Out compartilhado, é recomendável que o provedor do contrato seja o L3Out compartilhado e o EPG seja o consumidor do contrato. Este cenário será ilustrado nesta seção.

Não é recomendável fazer o oposto, que é L3Out consumindo um serviço fornecido por um EPG. O motivo disso tem a ver com a escalabilidade, já que para serviços compartilhados, as regras de zoneamento são instaladas apenas no VRF do consumidor. Os princípios de consumo e fornecimento denotam onde os fluxos de tráfego são iniciados. Com a aplicação da política de ingresso padrão, isso significa que a aplicação da política será aplicada no lado do consumidor e, mais especificamente, na folha de ingresso (folha não borda). Para que o leaf de entrada aplique a política, ele requer o pcTag do destino. Neste cenário, o destino é o EPG pcTag externo. A folha

de entrada executa a aplicação da política e encaminha os pacotes para a folha de borda. A folha de borda recebe o pacote em seu link de estrutura que executa uma pesquisa de rota (LPM) e encaminha o pacote para a adjacência do prefixo de destino.

A folha de borda, no entanto, NÃO executa nenhuma aplicação de política ao enviar tráfego para o EP de destino, nem faz isso no fluxo de tráfego de retorno de volta para o EP de origem.

Como resultado, somente o CAM de política da folha não BL de ingresso tem entradas instaladas (no VRF de consumidor) e o CAM de política do BL não é afetado.

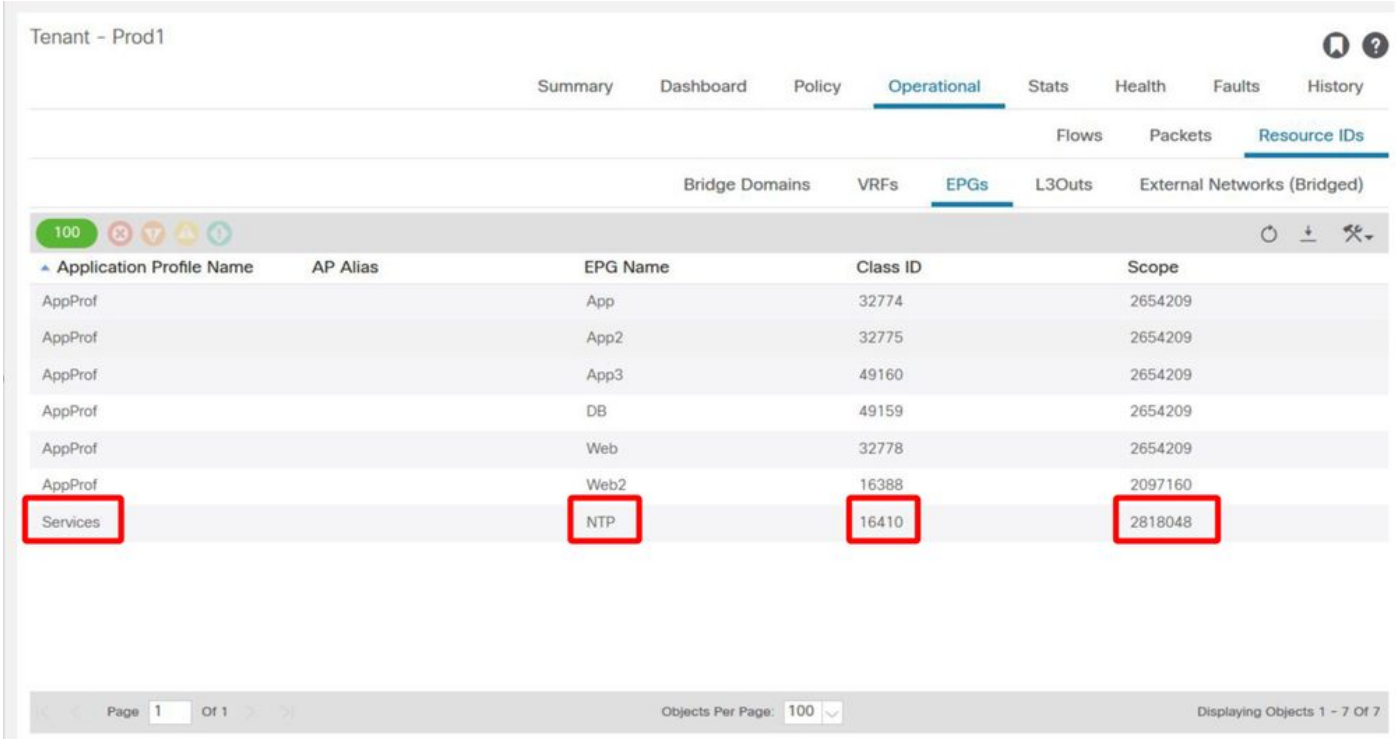
Troubleshooting de uma L3out Compartilhada

Fluxo de trabalho

1. Verificar EPG pcTag e VRF VNID/Scope para o EPG do consumidor

Com L3Out compartilhado, as regras de zoneamento são instaladas somente no VRF do consumidor. O provedor deve ter um pcTag global (abaixo de 16k) que permita que esse pcTag seja usado em todos os VRFs de consumidor. Em nosso cenário, o provedor é o EPG externo e terá um pcTag global. O EPG do consumidor terá um pcTag local como de costume.

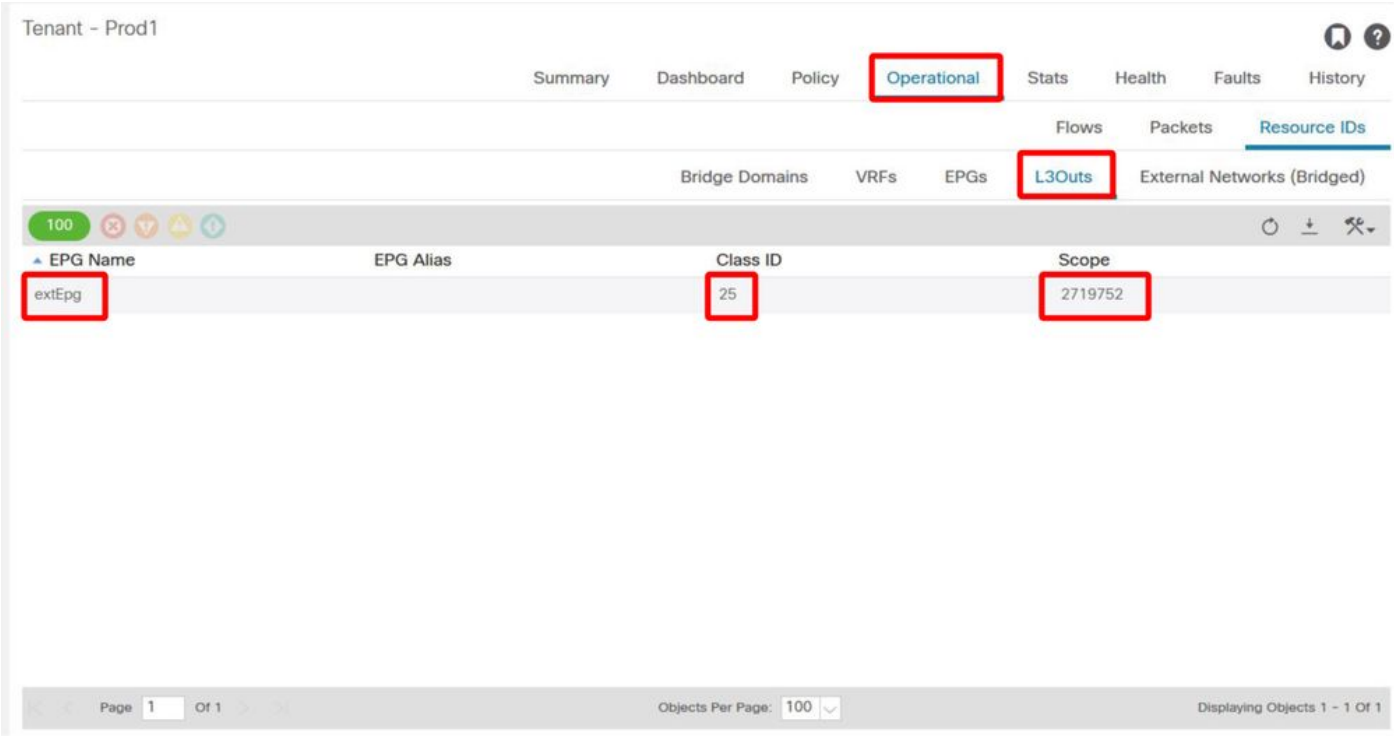
pcTag de EPG de consumidor



2. Verificar o pcTag e VRF VNID/Scope para o provedor L3Out EPG

Como observado na Etapa 1, o provedor L3Out EPG tem um pcTag de intervalo global como prefixos da L3Out que vazam para o VRF de consumidor. Como resultado, o L3Out EPG pcTag é necessário para não se sobrepor a pcTags no VRF de consumidor e, portanto, está dentro do intervalo global de pcTag.

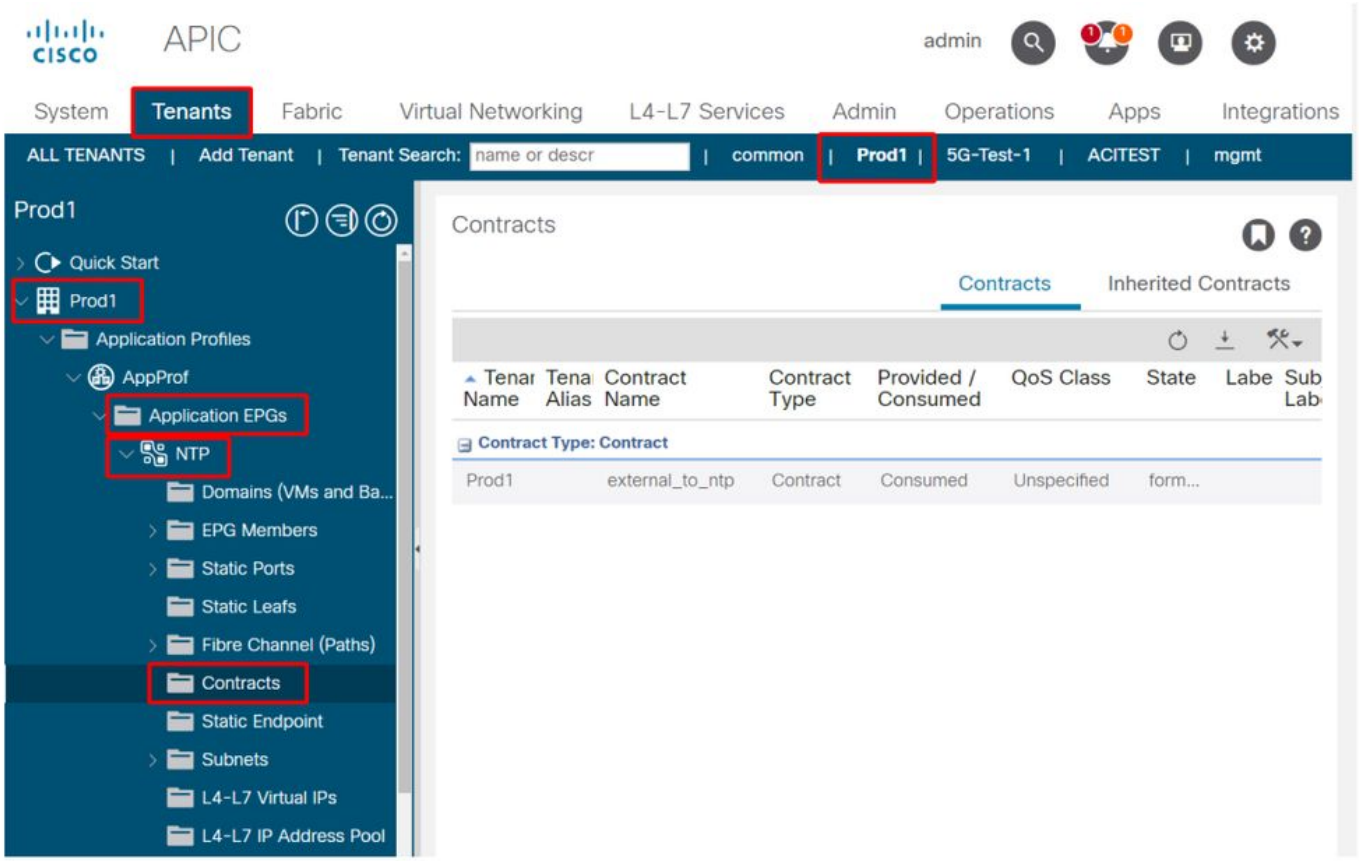
pcTag do EPG externo do provedor



3. Verifique se o EPG do consumidor tem um contrato com escopo de locatário importado ou um contrato global configurado

O NTP de EPG de consumidor com sub-rede definida no EPG/BD está consumindo o contrato com escopo 'locatário' ou 'global'

Contrato consumido pelo EPG



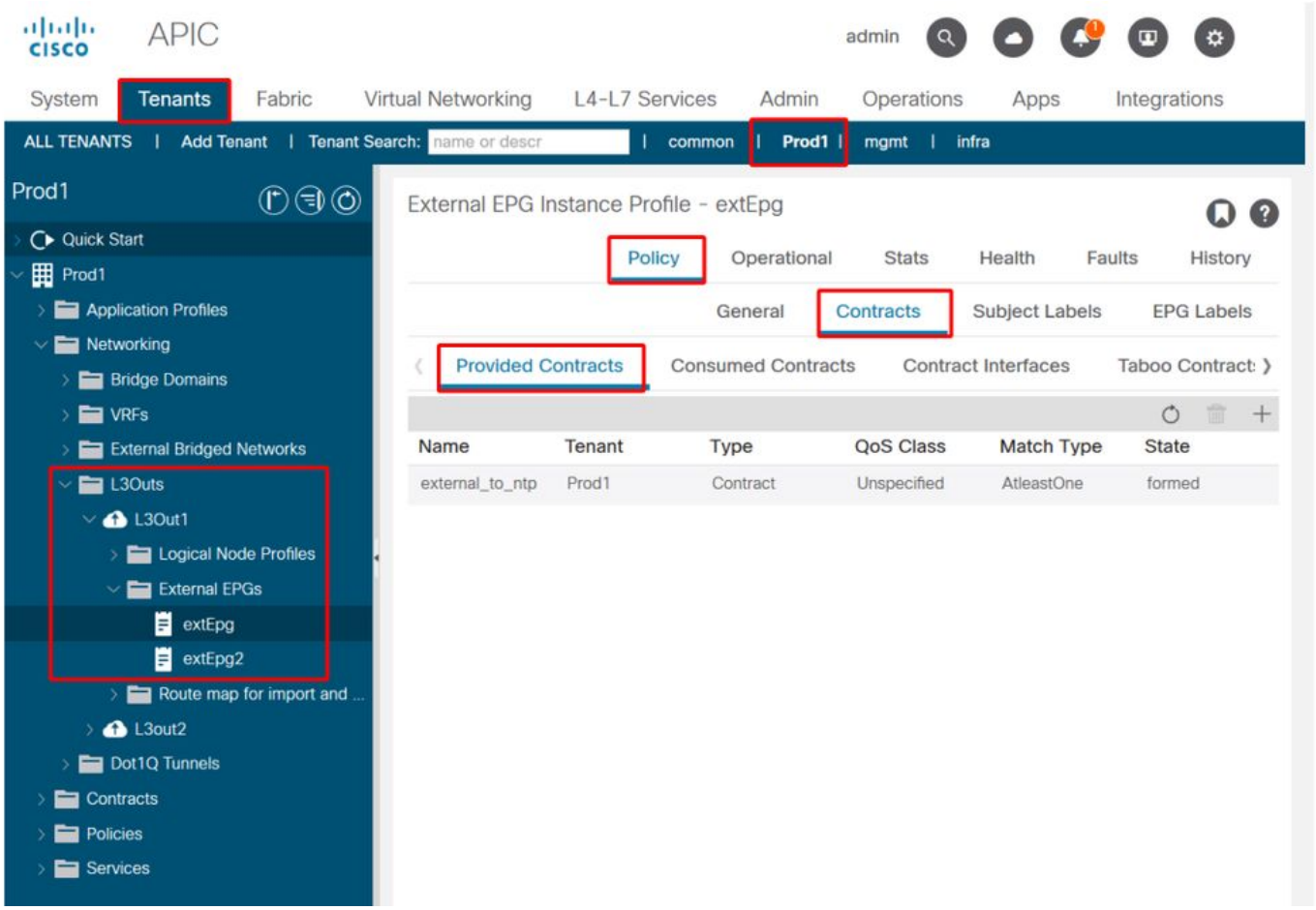
4. Verifique se o BD do EPG do consumidor tem uma sub-rede configurada com seu escopo definido como 'Compartilhado entre VRFs'

A sub-rede do EPG está configurada no domínio de bridge, mas deve ter o sinalizador 'shared between VRF' (shared between VRF) (para permitir vazamento roteado) e o sinalizador 'advertised externally' (para permitir anúncio em L3Out)

5. Verifique se o provedor L3Out EPG tem um contrato com escopo de locatário importado ou um contrato global configurado

O L3Out EPG deve ter um contrato com escopo de espaço ou um contrato global configurado como um contrato fornecido.

Contrato no provedor L3Out



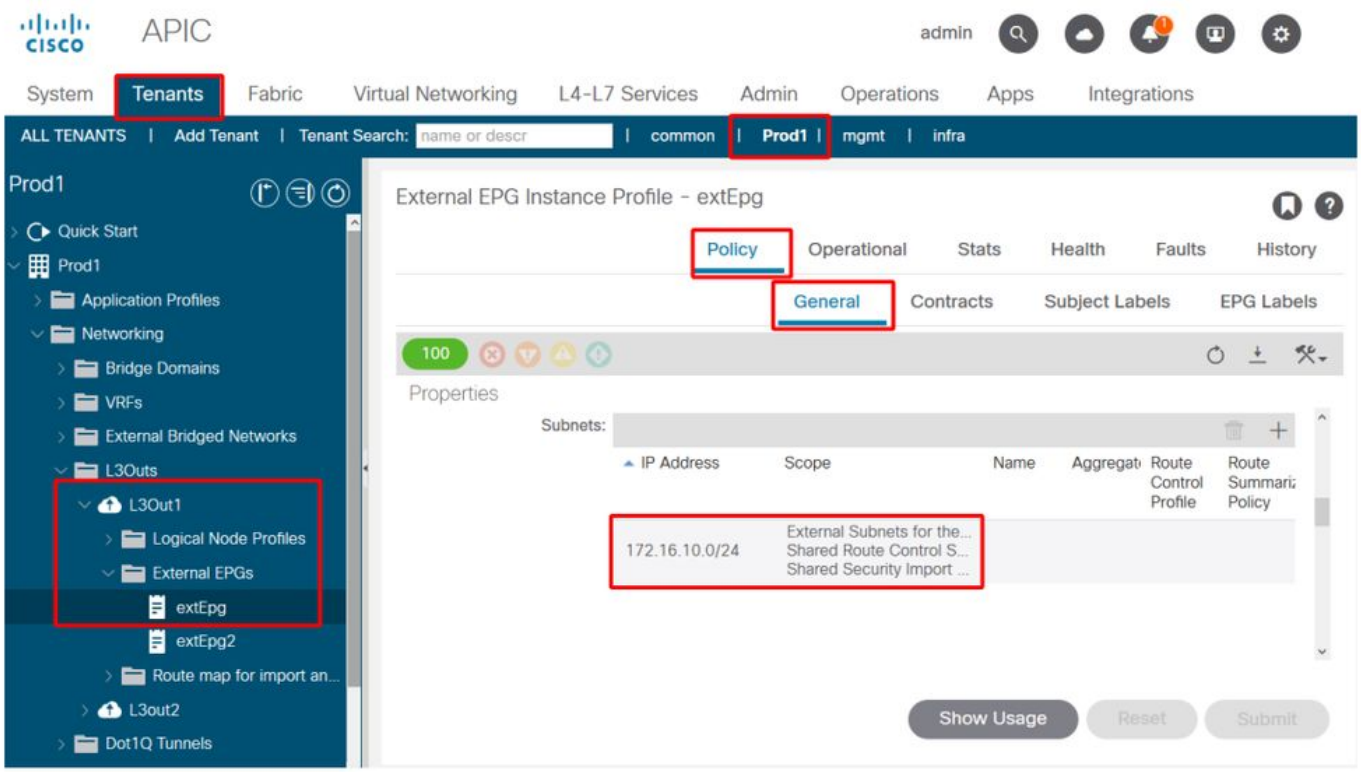
6. Verifique se o EPG de L3Out do provedor tem uma sub-rede configurada com os escopos necessários marcados

O provedor L3Out EPG deve ter o prefixo a ser vazado configurado com os seguintes escopos:

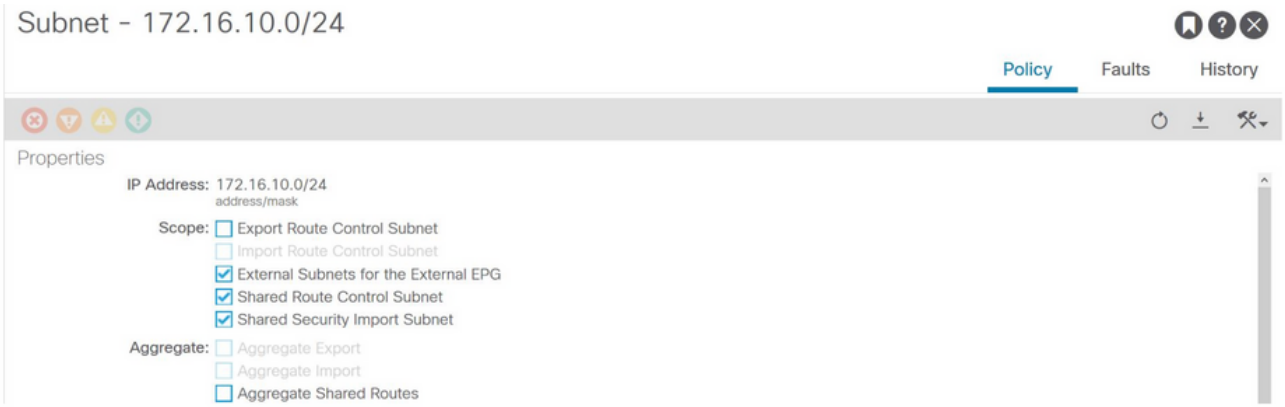
- Sub-redes externas para o EPG externo.
- Sub-rede de controle de rota compartilhada.
- Sub-rede de importação de segurança compartilhada.

Para obter mais detalhes sobre a flag de sub-rede em L3Out EPG, consulte o capítulo "Encaminhamento externo".

Configurações de sub-rede EPG externa



Configurações de sub-rede EPG externa expandidas



7. Verifique o pcTag da sub-rede L3Out EPG na não-BL para o VRF de consumidor

Quando o tráfego destinado à sub-rede EPG externa entra no não-BL, é realizada uma consulta no prefixo de destino para determinar o pcTag. Isso pode ser verificado usando o seguinte comando no não-BL.

Observe que essa saída é obtida no escopo do 2818048 VNI, que é o VRF VNID do consumidor. Olhando a tabela, o consumidor pode encontrar o pcTag do destino, mesmo que não esteja no mesmo VRF.

```
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
```

A saída acima mostra a combinação da sub-rede L3Out EPG e seu pcTag 25 global.

8. Verifique as regras de zoneamento programadas no não-BL para o VRF de consumidor

Use 'contract_parser.py' ou o comando 'show zoning-rule' e especifique o VRF.

As saídas do comando a seguir exibem duas regras de zoneamento instaladas para permitir o tráfego do 16410 pcTag local do EPG de consumidor para o pcTag global 25 do EPG de saída L3Out. Isso está no escopo 2818048, que é o escopo do VRF de consumidor.

```
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. Verifique as regras de zoneamento programadas no BL para o VRF do provedor

Use 'contract_parser.py' ou o comando 'show zoning-rule' e especifique o VRF. As saídas de comando a seguir mostram que **NÃO** há regras de zoneamento específicas no VRF do provedor conforme descrito várias vezes antes.

Está no escopo 2719752 qual é o escopo do VRF do provedor.

```
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]
```

Sobre esta tradução
A Cisco traduziu este documento com a ajuda de tecnologias de tradução automática e humana para oferecer conteúdo de suporte aos seus usuários no seu próprio idioma, independentemente da localização.

Observe que mesmo a melhor tradução automática não será tão precisa quanto as realizadas por um tradutor profissional.

A Cisco Systems, Inc. não se responsabiliza pela precisão destas traduções e recomenda que o documento original em inglês (link fornecido) seja sempre consultado.