패킷 색상 지정 기술 또는 플랫폼 카운터를 사용하여 패킷 삭제 트러블슈팅

목차

[소개](#)

[사전 요구 사항](#)

- [요구 사항](#)
- [사용되는 구성 요소](#)

[배경 정보](#)

[토폴로지](#)

[옵션 1. Flow-ID가 있는 ERSPAN 설정](#)

- [1단계. ESPAN 대상 설정](#)
- [2a 단계. SRC에 직접 연결된 트래픽에 대한 Span Source 생성](#)
- [2b단계. DST에 직접 연결된 트래픽에 대한 Span 소스 생성](#)
- [3단계. 빠른 Wireshark 분석](#)

[옵션 2. 플랫폼 카운터](#)

- [플랫폼 카운터 지우기](#)
- [패킷 크기가 작거나 0인 패킷 식별](#)
- [트래픽 흐름 추적](#)

[관련 정보](#)

소개

이 문서에서는 패킷 색상 지정 기술을 사용하여 네트워크 흐름을 추적하는 방법에 대해 설명합니다 .

사전 요구 사항

요구 사항

- ACI에 대한 기본 지식
- 엔드포인트 그룹 및 계약
- Wireshark 기본 지식

사용되는 구성 요소

이 문서는 특정 하드웨어 및 소프트웨어 버전으로 제한되지 않습니다.

사용된 장치:

- 버전 5.3(2)을 실행하는 Cisco ACI

- 대상 범위
- Gen2 스위치

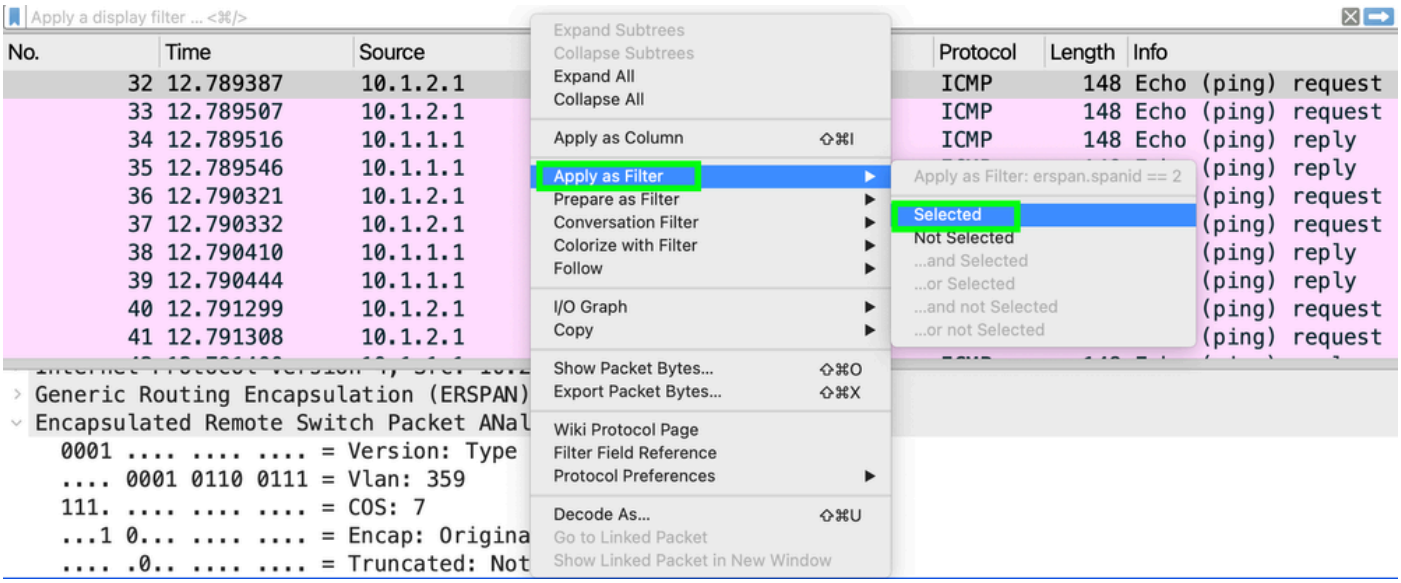
이 문서의 정보는 특정 랩 환경의 디바이스를 토대로 작성되었습니다. 이 문서에 사용된 모든 디바이스는 초기화된(기본) 컨피그레이션으로 시작되었습니다. 현재 네트워크가 작동 중인 경우 모든 명령의 잠재적인 영향을 미리 숙지하시기 바랍니다.

배경 정보

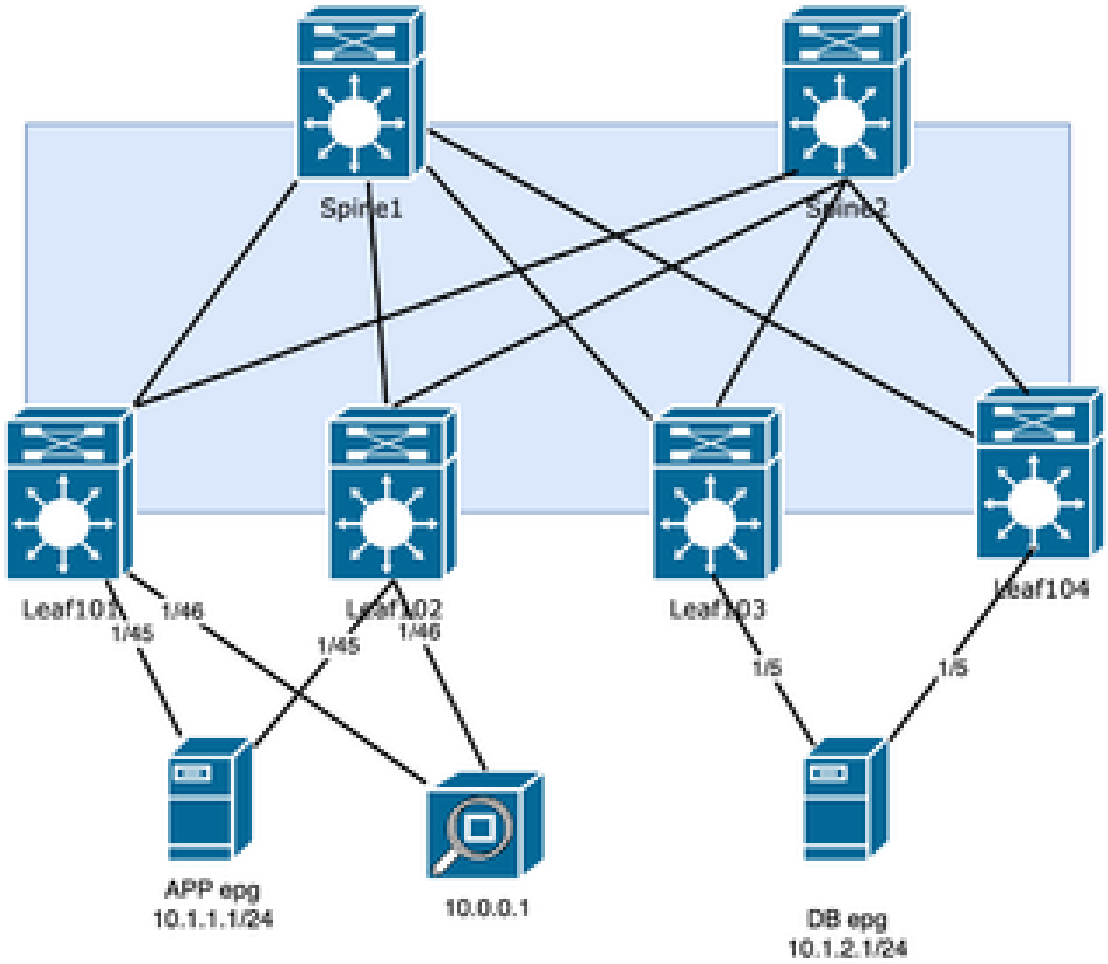
Wireshark에서 필터를 만드는 방법.

캡처를 엽니다. Encapsulated Remote Switch Packet(캡슐화된 원격 스위치 패킷) 내부의 프레임을 사용하여 SpanID 라인을 선택하고 마우스 오른쪽 버튼을 클릭합니다.

그림에 표시된 대로 Apply as Filter(필터로 적용) > Selected(선택)를 선택합니다.



토폴로지



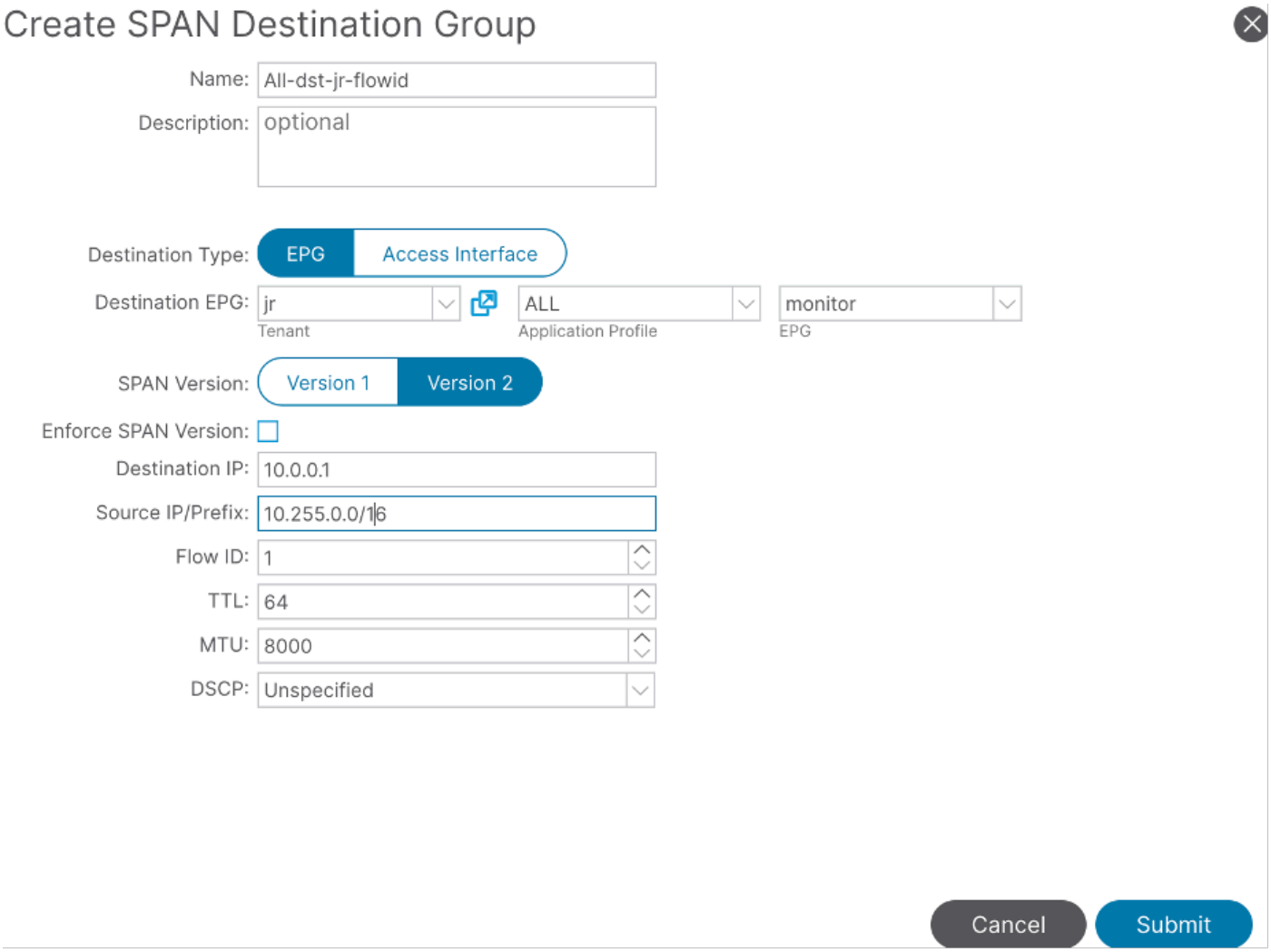
옵션 1. Flow-ID가 있는 ERSPAN 설정

목적지 서버에서 모든 트래픽을 처리할 수 있는 경우, ERSPAN 헤더에는 Flow ID를 정의하는 옵션이 포함됩니다. 이 Flow ID는 패브릭으로 들어오는 트래픽을 식별하도록 구성할 수 있으며, 나가는 트래픽에 대해 다른 Flow ID를 설정할 수 있습니다.

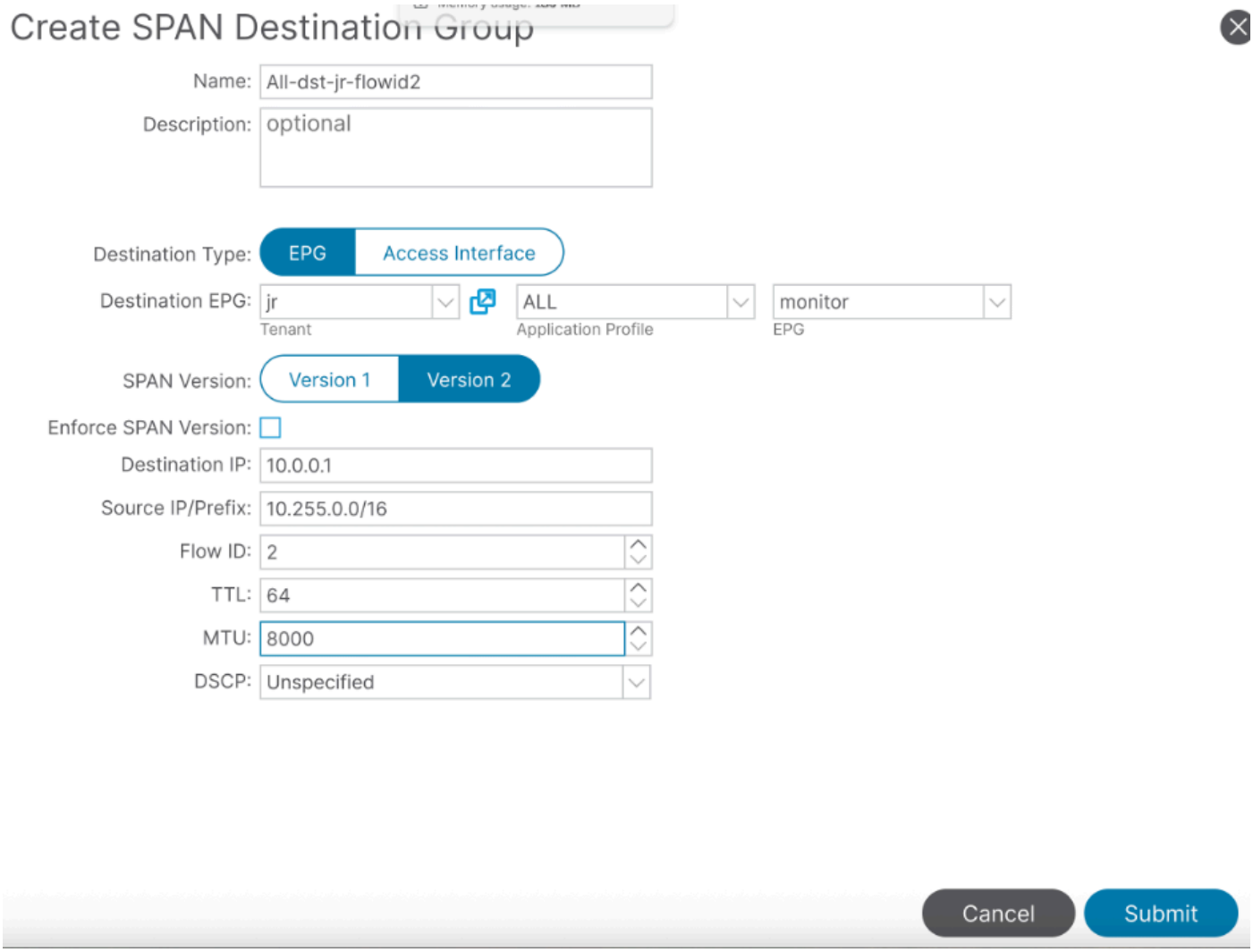
1단계. ESPAN 대상 설정

하나의 대상 그룹은 flow-id가 1이 됩니다.

Fabric(패브릭) > Access Policies(액세스 정책) > Policies(정책) > Troubleshooting(문제 해결) > SPAN > SPAN Destination Groups(SPAN 대상 그룹)에서

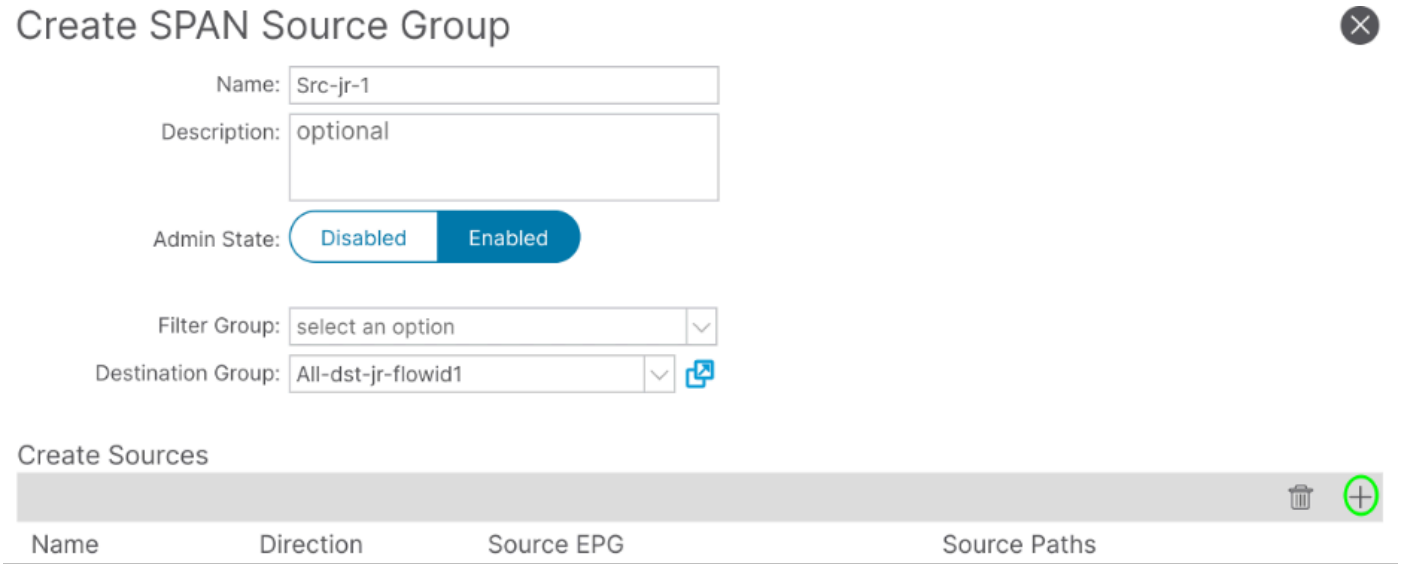


두 번째 대상 그룹에서 flow-id 2를 구성합니다.



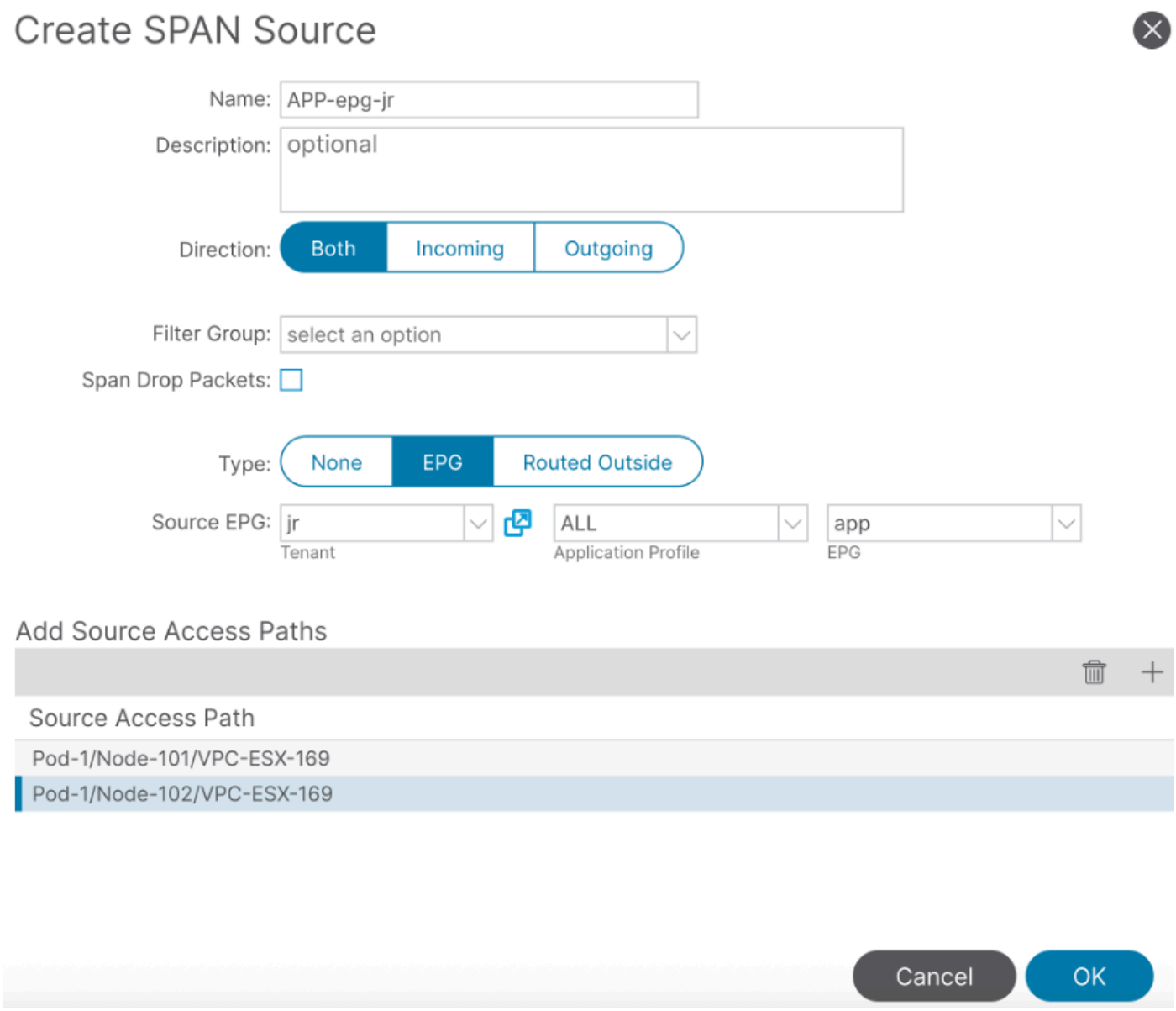
2a 단계. SRC에 직접 연결된 트래픽에 대한 Span 소스 생성

Fabric(패브릭) > Access Policies(액세스 정책) > Policies(정책) > Troubleshooting(문제 해결) > SPAN > SPAN Source Groups(SPAN 소스 그룹)에서



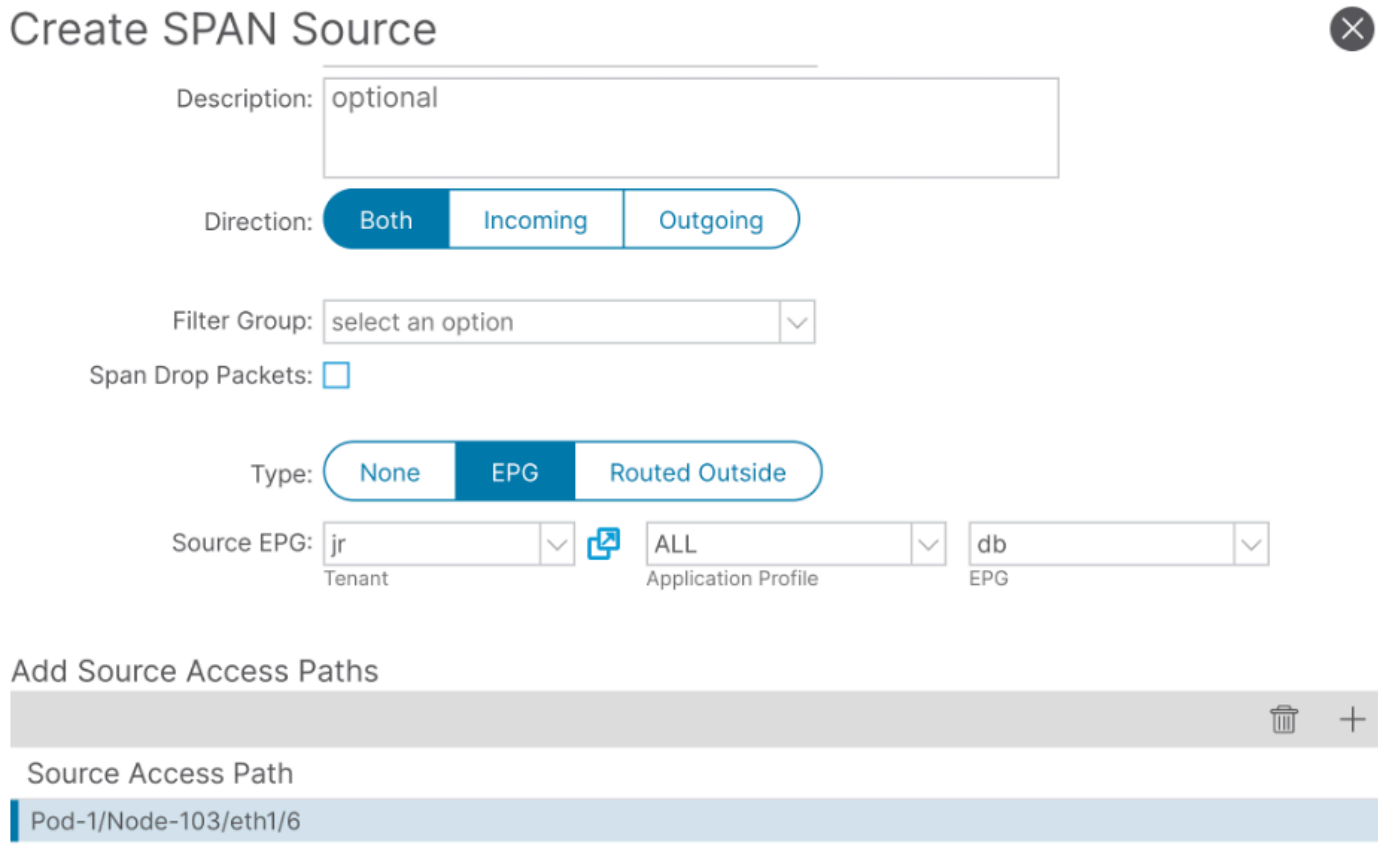
경로와 EPG를 추가하여 트래픽을 더 필터링합니다. 실습의 예는 Tenant jr Application Profile ALL

및 EPG 앱입니다.

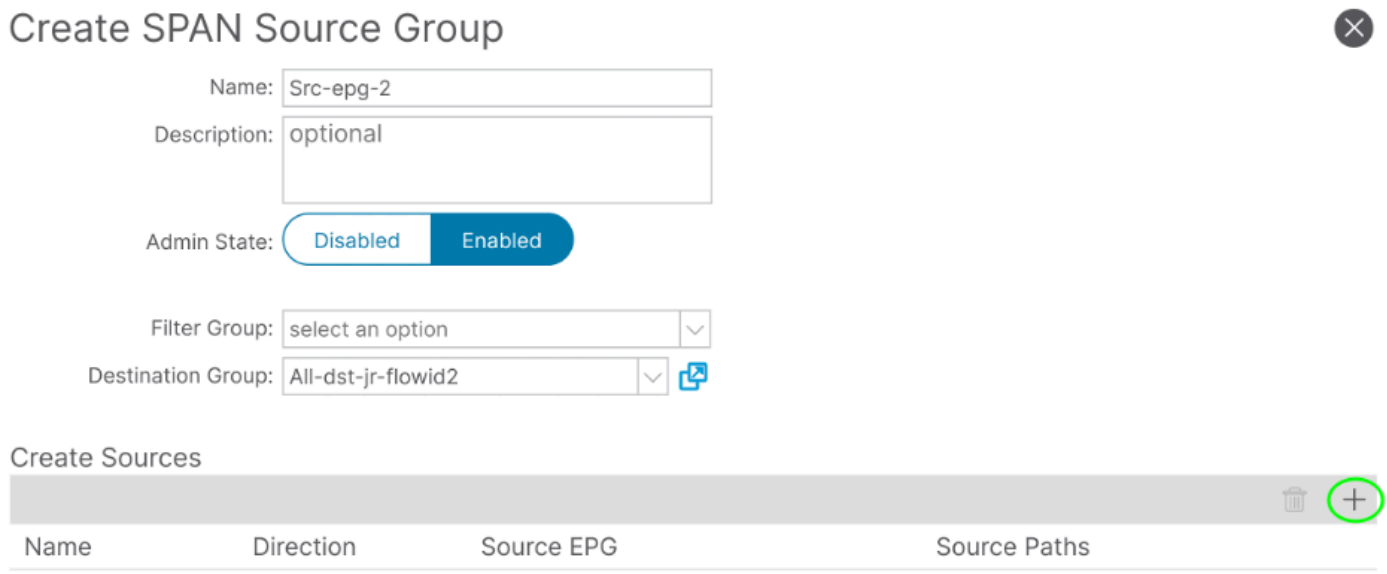


2b단계. DST에 직접 연결된 트래픽에 대한 Span 소스 생성

Fabric(패브릭) > Access Policies(액세스 정책) > Policies(정책) > Troubleshooting(문제 해결) > SPAN > SPAN Source Groups(SPAN 소스 그룹)에서



경로뿐만 아니라 EPG DB도 추가하여 트래픽을 더 많이 필터링합니다.



3단계. 빠른 Wireshark 분석

이 예에서는 ICMP 요청 패킷의 수가 ICMP 응답 패킷의 수와 일치하는지 확인하여 ACI 패브릭 내에 패킷 드랍이 없음을 확인합니다.

wireshark에서 캡처를 열어 SRC 및 DST IP와 함께 구성된 SPAN ID /Flow-ID를 사용하여 필터를 생성합니다.

<#root>

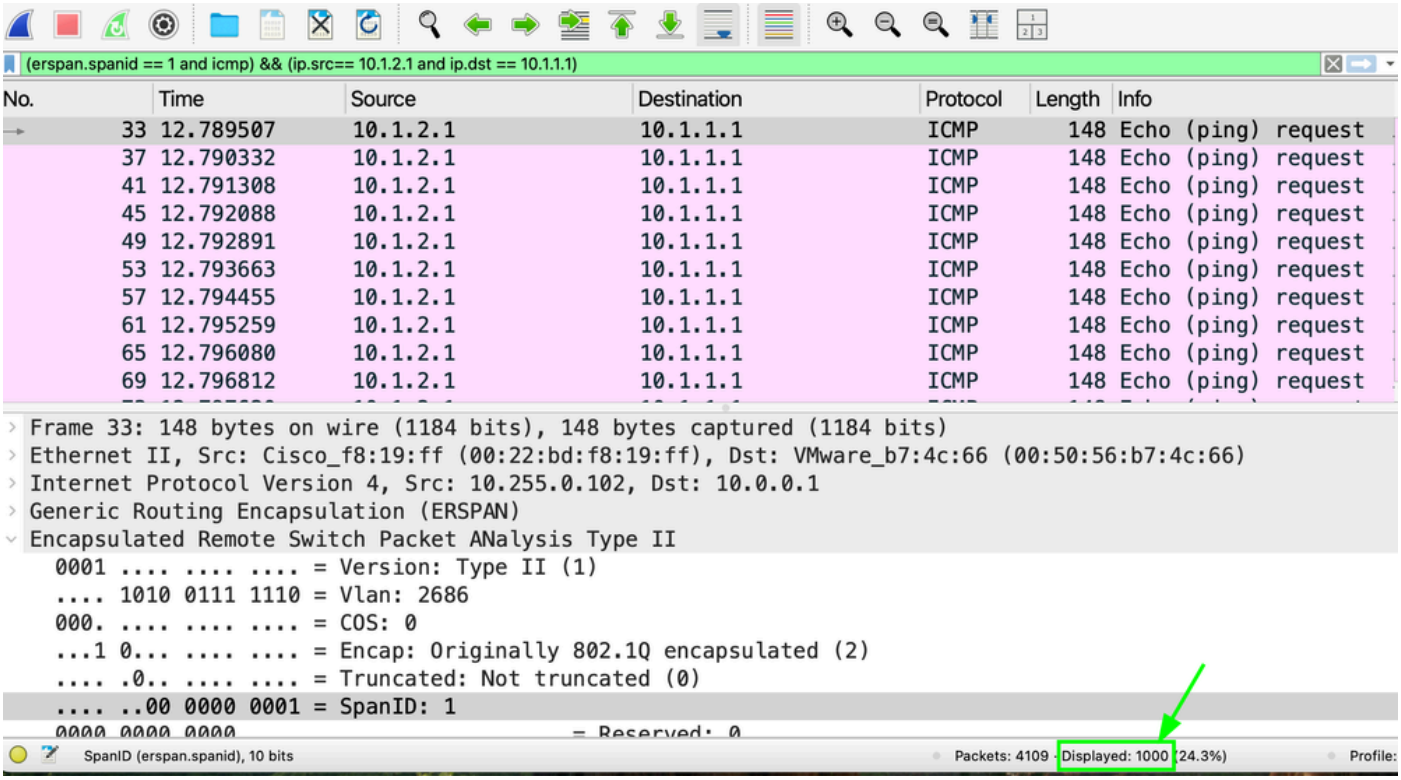
```
(erspan.spanid ==


        and


        ) && (ip.src==


          and ip.dst ==


          )
```

랩 테스트 플로우에 사용되는 필터:

<#root>

```
(erspan.spanid == 1 and icmp) && (ip.src== 10.1.2.1 and ip.dst == 10.1.1.1)
```

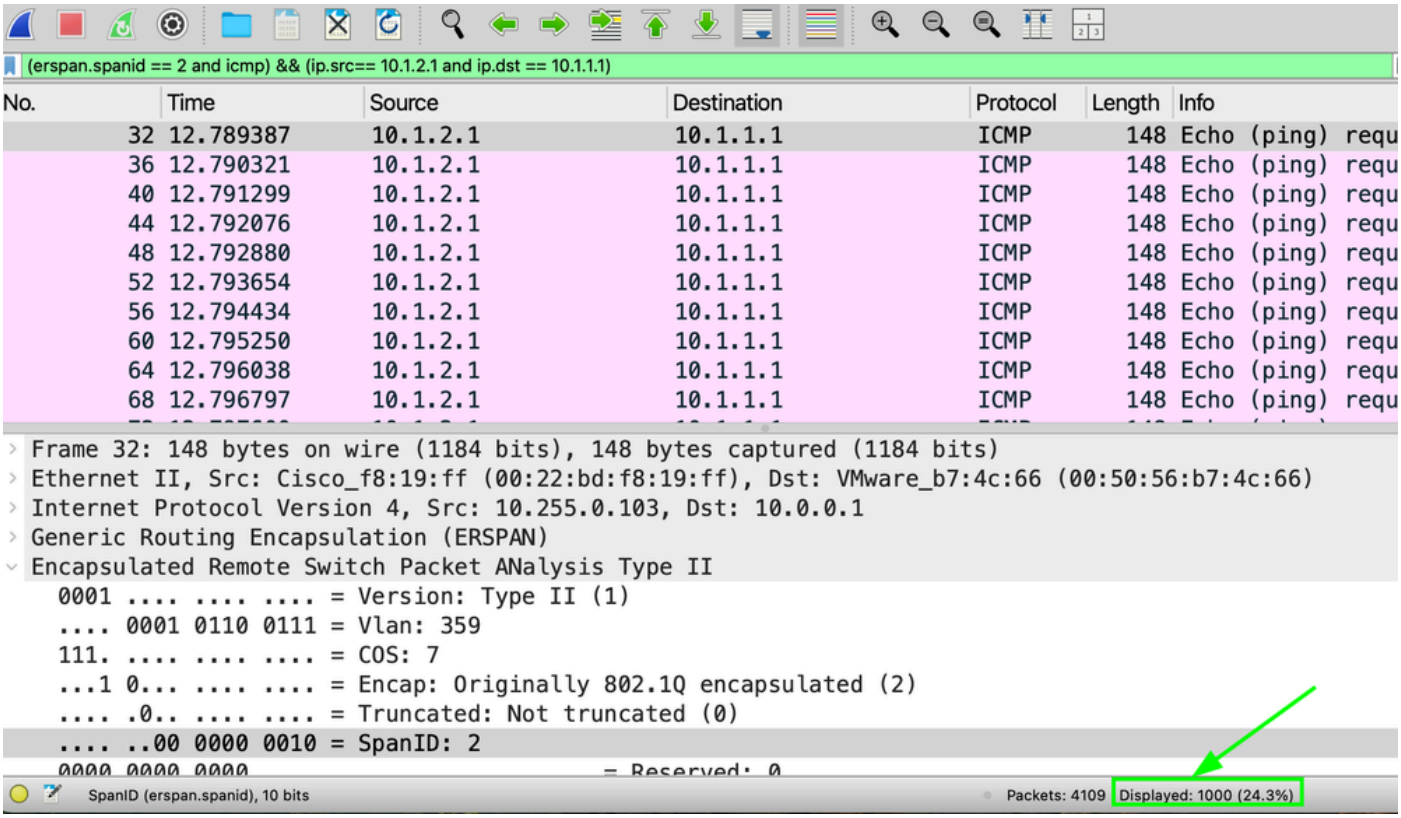
Displayed packet is the same amount as sent(표시된 패킷이 전송된 금액과 동일한지 확인):



다음 SPAN ID의 금액은 같아야 합니다. 그렇지 않은 경우 패킷이 패브릭 내부에 드롭되었습니다.

필터:

```
(erspan.spanid == 2 and icmp) && (ip.src== 10.1.2.1 and ip.dst == 10.1.1.1)
```



옵션 2. 플랫폼 카운터

이 방법은 Nexus가 패킷 크기가 다른 개별 인터페이스의 성능을 추적한다는 점을 이용하지만, 최소한 큐의 트래픽 양이 적어야 합니다(0은 아님).

플랫폼 카운터 지우기

개별 스위치로 이동하여 디바이스에 연결되는 개별 인터페이스를 지웁니다.

<#root>

Switch#

vsh_lc -c “clear platform internal counters port

”

<#root>

LEAF3#

vsh_lc -c “clear platform internal counters port 6”

LEAF1#

vsh_lc -c “clear platform internal counters port 45”

LEAF2#

vsh_lc -c “clear platform internal counters port 45”

패킷 크기가 작거나 0인 패킷 식별

RX 및 TX에 대한 모든 Leaf에서 카운터가 없을 수 있는 패킷 크기를 찾습니다.

<#root>

vsh_lc -c ‘show platform internal counters port

```
' | grep X_PKT
```

다음 예에서는 패킷 크기가 512보다 크고 1024보다 작습니다.

<#root>

LEAF101#

```
vsh_lc -c "show platform internal counters port 45 " | grep X_PKT

                RX_PKTOK           1187
                RX_PKTTOTAL        1187
                RX_PKT_LT64           0
                RX_PKT_64             0
                RX_PKT_65          1179
                RX_PKT_128            8
                RX_PKT_256            0

                RX_PKT_512            0 <<


                RX_PKT_1024           0
                RX_PKT_1519           0
                RX_PKT_2048           0
                RX_PKT_4096           7
                RX_PKT_8192          43
                RX_PKT_GT9216         0
                TX_PKTOK           3865
                TX_PKTTOTAL        3865
                TX_PKT_LT64           0
                TX_PKT_64             0
                TX_PKT_65          3842
                TX_PKT_128           17
                TX_PKT_256            6

                TX_PKT_512            0 <<


                TX_PKT_1024          10
                TX_PKT_1519           3
                TX_PKT_2048         662
                TX_PKT_4096           0
                TX_PKT_8192           0
                TX_PKT_GT9216         0
```

이 단계는 패킷이 전달되는 링크에서 수행해야 합니다.

트래픽 흐름 추적

서버 10.1.2.1에서 1000개의 패킷이 520의 패킷 크기로 전송됩니다.

트래픽이 RX에서 시작되는 Leaf 103 인터페이스 1/6에서 확인합니다.

<#root>

MXS2-LF103#

vsh_lc -c "show platform internal counters port 6 " | grep X_PKT_512

```
                RX_PKT_512         1000
                TX_PKT_512          647
```

1000 패킷 RX이지만 647개만 회신으로 전송되었습니다.

다음 단계는 다른 서버의 발신 인터페이스를 확인하는 것입니다.

Leaf102의 경우

<#root>

MXS2-LF102#

vsh_lc -c "show platform internal counters port 45 " | grep X_PKT_512

```
                RX_PKT_512            0
                TX_PKT_512         1000
```

패브릭에서 요청을 삭제하지 않았습니다.

리프 101의 경우, RX 패킷 647이며 ACI에 의한 패킷 TX의 동일한 양입니다.

<#root>

MXS2-LF101#

vsh_lc -c "show platform internal counters port 45 " | grep X_PKT_512

```
                RX_PKT_512         647
                TX_PKT_512           0
```

관련 정보

ACI Intra-Fabric Forwarding 문제 해결 - 간헐적 삭제

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.