

Nexus 플랫폼의 인터페이스 카운터로 패킷 삭제 트러블슈팅

목차

- [소개](#)
 - [사전 요구 사항](#)
 - [요구 사항](#)
 - [사용되는 구성 요소](#)
 - [토폴로지](#)
 - [배경](#)
 - [인터페이스 식별](#)
 - [N9K1의 경로](#)
 - [N9K2의 경로](#)
 - [N9K3의 경로](#)
 - [패킷 크기 식별](#)
 - [테스트 수행](#)
 - [ICMP 요청 확인](#)
 - [ICMP 응답 확인](#)
-

소개

이 문서에서는 Nexus 인터페이스 카운터를 사용하여 패킷 손실을 해결하는 방법에 대해 설명합니다.

사전 요구 사항

요구 사항

다음 주제에 대한 지식을 보유하고 있으면 유용합니다.

- NXOS 플랫폼

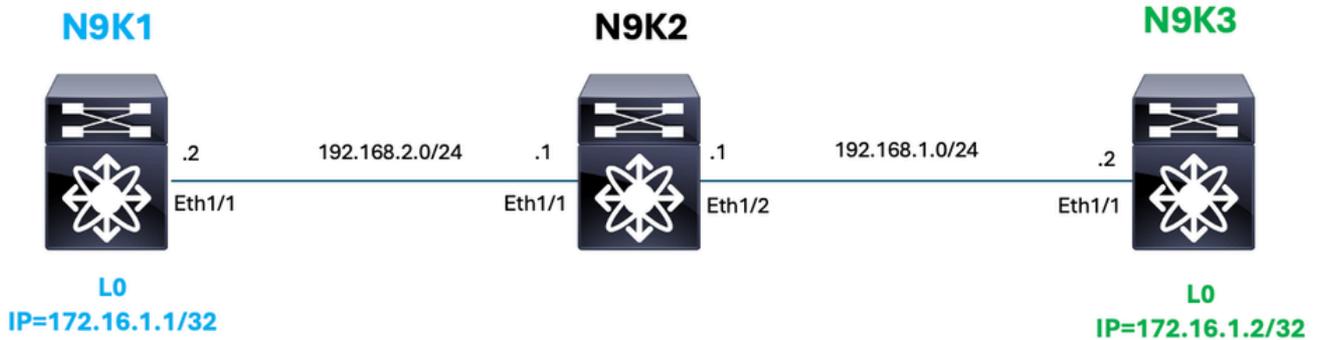
사용되는 구성 요소

이름	플랫폼	버전
N9K1	N9K-C93108TC-EX	9.3(10)
N9K2	N9K-C93108TC-EX	9.3(10)
N9K3	N9K-C93108TC-EX	9.3(10)

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.

토폴로지



배경

특정 환경에서는 ELAM 또는 SPAN과 같은 기존 패킷 캡처 방법으로 네트워크 문제를 진단할 수 없습니다. 그러나 Nexus 인터페이스 패킷 카운터는 패킷 삭제 트러블슈팅을 위한 유용한 대안을 제공합니다. 네트워크 구성에 따라 특정 카운터의 가용성이 달라질 수 있으므로 이러한 트러블슈팅 방법은 보편적으로 적용할 수 없습니다.

이 예에서는 Nexus 인터페이스 카운터를 사용하여 N9K1(172.16.1.1)과 N9K3(172.16.1.2) 의 루프백 인터페이스 간의 연결 문제를 해결하는 방법을 보여 줍니다.

인터페이스 식별

이 예시 명령에 대해 이 인터페이스를 식별하려면 각 디바이스에 대해 인그레스 및 이그레스 인터페이스를 식별해야 합니다. show ip route is utilized.

N9K1의 경로

```
<#root>
```

```
N9K1
```

```
# sh ip route 172.16.1.2
```

```
<Snipped>
```

```
172.16.1.2/32, ubest/mbest: 1/0
```

```
  *via 192.168.2.1,
```

```
Eth1/1
```

```
, [1/0], static
```

nexus N9K1의 경우 인터페이스 Eth1/1이 사용됩니다.

N9K2의 경로

```
<#root>
```

```
N9K2
```

```
# sh ip route 172.16.1.1  
<Snipped>  
172.16.1.1/32, ubest/mbest: 1/0 time  
  *via 192.168.2.2,
```

```
Eth1/1
```

```
, [1/0], static
```

```
N9K2
```

```
# sh ip route 172.16.1.2  
<Snipped>  
172.16.1.2/32, ubest/mbest: 1/0 time  
  *via 192.168.1.2,
```

```
Eth1/2
```

```
, [1/0], static
```

nexus N9K1의 경우, 인터페이스 Eth1/1 및 Eth1/2가 사용됩니다.

N9K3의 경로

```
<#root>
```

```
N9K3
```

```
# sh ip route 172.16.1.1  
<Snipped>  
172.16.1.1/32, ubest/mbest: 1/0 time  
  *via 192.168.1.1,
```

```
Eth1/1
```

```
, [1/0], static
```

nexus N9K1의 경우 인터페이스 Eth1/1이 사용됩니다.

패킷 크기 식별

인터페이스 카운터를 사용하여 핫 패킷 삭제를 트러블슈팅하려면 증가하지 않는 카운터를 식별해야 합니다.

다음 예에서는 명령 sh 인터페이스 e1/1 카운터가 두 번 실행되었으며, 512~1023바이트의 카운터

패킷이 RX 및 TX에서 증가하지 않았음을 확인할 수 있습니다.

이 프로세스는 소스와 대상 사이의 모든 관련 디바이스에서 수행되어야 합니다.

<#root>

```
N9K1# sh interface e1/1 counters detailed
```

```
Ethernet1/1
```

```
Rx Packets: 31774
Rx Unicast Packets: 8419
Rx Multicast Packets: 23784
Rx Broadcast Packets: 3
Rx Bytes: 8115383
Rx Packets from 0 to 64 bytes: 322
Rx Packets from 65 to 127 bytes: 22822
Rx Packets from 128 to 255 bytes: 3393
Rx Packets from 256 to 511 bytes: 1652

Rx Packets from 512 to 1023 bytes: 63

Rx Packets from 1024 to 1518 bytes: 3522
Tx Packets: 26430
Tx Unicast Packets: 7351
Tx Multicast Packets: 19509
Tx Broadcast Packets: 2
Tx Bytes: 5114894
Tx Packets from 0 to 64 bytes: 90
Tx Packets from 65 to 127 bytes: 20724
Tx Packets from 128 to 255 bytes: 2243
Tx Packets from 256 to 511 bytes: 1642

Tx Packets from 512 to 1023 bytes: 10

Tx Packets from 1024 to 1518 bytes: 1766
```

```
N9K1# sh interface e1/1 counters detailed
```

```
Ethernet1/1
```

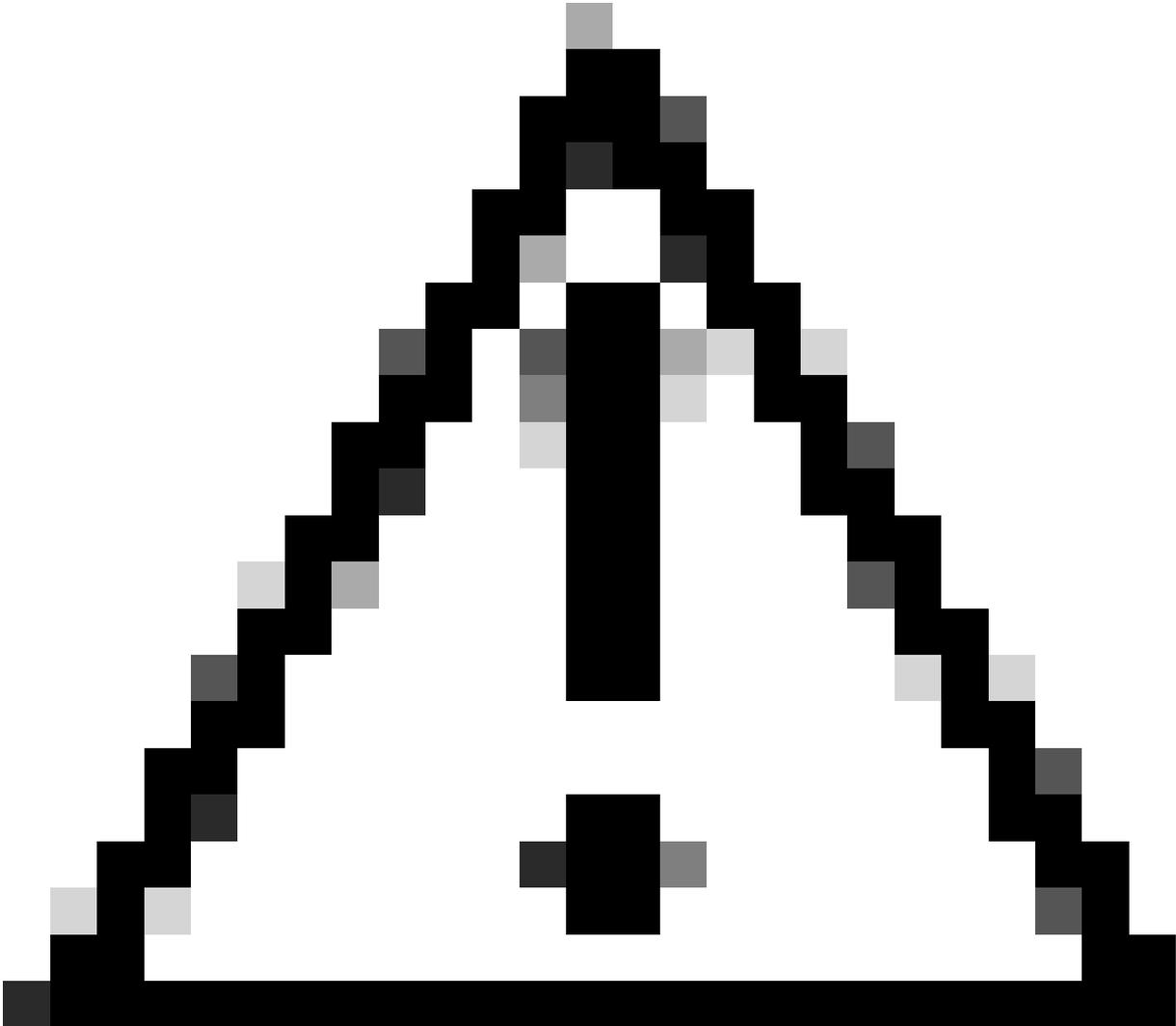
```
Rx Packets: 31821
Rx Unicast Packets: 8437
Rx Multicast Packets: 23817
Rx Broadcast Packets: 3
Rx Bytes: 8125733
Rx Packets from 0 to 64 bytes: 329
Rx Packets from 65 to 127 bytes: 22878
Rx Packets from 128 to 255 bytes: 3468
Rx Packets from 256 to 511 bytes: 1670

Rx Packets from 512 to 1023 bytes: 63

Rx Packets from 1024 to 1518 bytes: 3544
Tx Packets: 26467
Tx Unicast Packets: 7367
Tx Multicast Packets: 19534
Tx Broadcast Packets: 2
Tx Bytes: 5121572
Tx Packets from 0 to 64 bytes: 95
Tx Packets from 65 to 127 bytes: 20768
Tx Packets from 128 to 255 bytes: 2290
Tx Packets from 256 to 511 bytes: 1657
```

Tx Packets from 512 to 1023 bytes: 10

Tx Packets from 1024 to 1518 bytes: 1798



주의: 프로덕션 환경에서는 어떤 카운터가 증가하지 않는지 확인하기 위해 인터페이스 카운터를 지울 수 있습니다. MTU가 최대값으로 설정된 인터페이스의 경우 1518보다 큰 카운터를 찾을 수 있습니다. 특정 크기의 패킷이 Nexus를 통과하지 못하면 카운터가 표시되지 않습니다.

테스트 수행

이 테스트에서는 제어된 환경을 사용하므로 모든 디바이스에서 1024~1518의 카운터 패킷이 사용 됩니다. 테스트 전에 모든 인터페이스의 카운터가 지워집니다.

<#root>

N9K1

```
# clear counters interface
e1/1
N9K2
# clear counters interface
e1/1-2
N9K3
# clear counters interface
e1/1
```

모든 Nexus에서 다음 명령을 실행하여 원하는 패킷 크기의 트래픽이 Nexus를 통과하지 않는지 확인할 수 있습니다. 기대는 아무것도 보지 않는 것입니다.

```
<#root>
N9K1
# sh int
e1/1
cou detailed | i i " 1024 to 1518"
N9K2
# sh int
e1/1-2
  cou detailed | i i " 1024 to 1518"
N9K3
# sh int
e1/1
  cou detailed | i i " 1024 to 1518"
```

모든 카운터가 명확하므로 ping을 생성하여 DF-BIT가 설정된 1024-1518 사이의 패킷 크기를 지정할 수 있습니다.

```
<#root>
N9K1
# ping
172.16.1.2
  source 172.16.1.1 packet-size 1050 df-bit
PING 172.16.1.2(172.16.1.2) from 172.16.1.1: 1050 data bytes
```

```

1058 bytes from 172.16.1.2: icmp_seq=0 ttl=254 time=1.102 ms
1058 bytes from 172.16.1.2: icmp_seq=1 ttl=254 time=0.668 ms
1058 bytes from 172.16.1.2: icmp_seq=2 ttl=254 time=0.644 ms
1058 bytes from 172.16.1.2: icmp_seq=3 ttl=254 time=0.626 ms
1058 bytes from 172.16.1.2: icmp_seq=4 ttl=254 time=0.631 ms

```

```

--- 172.16.1.2 ping statistics ---
5 packets transmitted, 5 packets received, 0.00% packet loss
round-trip min/avg/max = 0.626/0.734/1.102 ms

```

ICMP 요청 확인

다음 예에서는 N9K1에서 N9K3으로의 ICMP 요청에 대해 관련 디바이스에서 TX/RX 방향으로 카운터가 증가하는 방식을 관찰할 수 있습니다.

N9K1	N9K2
<pre> <#root> N9K1# sh int e1/1 cou detailed i i " 1024 to 1518" 5 Rx Packets from 1024 to 1518 bytes: 0 Tx Packets from 1024 to 1518 bytes: 5 </pre>	<pre> <#root> N9K2# sh int e1/1 cou detailed i i " 1024 to 1518" 5 Rx Packets from 1024 to 1518 bytes: Tx Packets from 1024 to 1518 bytes: 0 N9K2# sh int e1/2 cou detailed i i " 1024 to 1518" 5 Rx Packets from 1024 to 1518 bytes: 0 Tx Packets from 1024 to 1518 bytes: 5 </pre>
<p>N9K1이 인터페이스 e1/1에서 5개의 패킷을 전송했음을 알 수 있습니다</p>	<p>N9K2는 인터페이스 e1/1에서 5개의 패킷을 수신하고 인터페이스 e1/2에서 5개의 패킷을 전송했음을 알 수 있습니다</p>

ICMP 응답 확인

ICMP 요청 패치가 확인되면 ICMP 응답을 검토할 수 있습니다.

다음 예에서는 N9K3에서 N9K1로의 ICMP 회신에 대해 관련 디바이스에서 TX/RX 방향으로 카운터가 어떻게 증가하는지 관찰할 수 있습니다

N9K1	N9K2
<pre> <#root> N9K1# sh int e1/1 cou detailed i i " 1024 to 1518" Rx Packets from 1024 to 1518 bytes: 5 Tx Packets from 1024 to 1518 bytes: 5 </pre>	<pre> <#root> N9K2# sh int e1/1 cou detailed i i " 1024 to 1518" Rx Packets from 1024 to 1518 bytes: 5 Tx Packets from 1024 to 1518 bytes: 5 N9K2# sh int e1/2 cou detailed i i " 1024 to 1518" Rx Packets from 1024 to 1518 bytes: 5 Tx Packets from 1024 to 1518 bytes: 5 </pre>
<p>N9K1은 인터페이스 e1/1에서 5개의 패킷을 수신함을 알 수 있습니다</p>	<p>N9K2는 인터페이스 e1/1에서 5개의 패킷을 전송하고 인터페이스 e1/2에서 5개의 패킷을 수신했음을 알 수 있습니다</p>

이 테스트를 통해 트래픽이 3개의 스위치에서 올바르게 이동했음을 확인할 수 있습니다. Nexus 중 하나에 카운터에서 불일치가 있는 경우, 트래픽을 삭제할 수 있는 RX 또는 TX입니다.

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.