

使用Nexus平台上的介面計數器排除資料包丟棄故障

目錄

- [簡介](#)
- [必要條件](#)
- [需求](#)
- [採用元件](#)
- [拓撲](#)
- [背景](#)
- [標識介面](#)
 - [N9K1中的路由](#)
 - [N9K2中的路由](#)
 - [N9K3中的路由](#)
- [標識資料包大小](#)
 - [執行測試](#)
 - [驗證ICMP請求](#)
 - [驗證ICMP應答](#)

簡介

本文說明如何使用Nexus介面計數器對封包遺失進行疑難排解。

必要條件

需求

思科建議您瞭解以下主題：

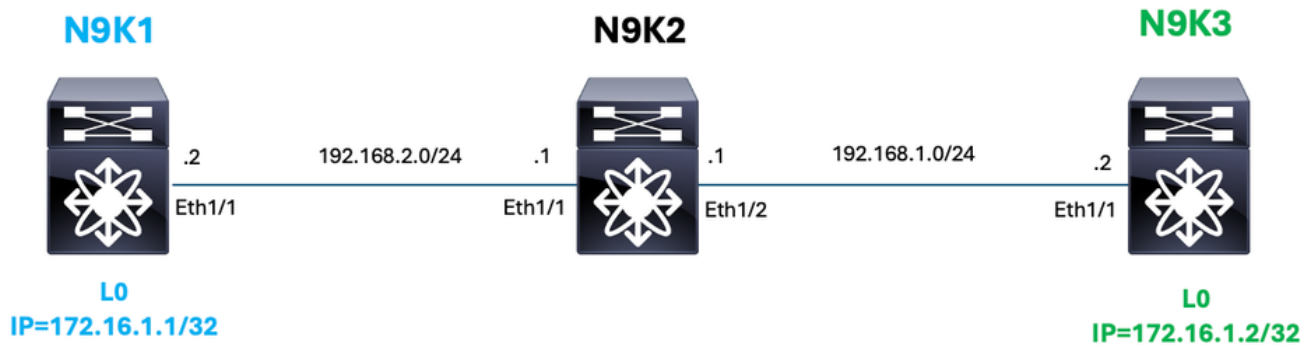
- NXOS平台

採用元件

名稱	平台	版本
N9K1	N9K-C93108TC-EX	9.3(10)
N9K2	N9K-C93108TC-EX	9.3(10)
N9K3	N9K-C93108TC-EX	9.3(10)

本文中的資訊是根據特定實驗室環境內的裝置所建立。文中使用到的所有裝置皆從已清除（預設）的組態來啟動。如果您的網路運作中，請確保您瞭解任何指令可能造成的影響。

拓撲



背景

在某些環境中，傳統的資料包捕獲方法（如ELAM或SPAN）不能作為診斷網路問題的可行選項。但是，Nexus介面資料包計數器為排除資料包丟棄故障提供了有價值的替代方案。必須注意的是，特定計數器的可用性可能會因網路配置而異，因此這種故障排除方法不能普遍適用。

在本示例中，演示了如何使用Nexus介面計數器來排除N9K1(172.16.1.1)和N9K3(172.16.1.2)的環回介面之間的連線問題。

標識介面

對於每個裝置，必須標識入口和出口介面，以便為此示例命令標識此介面：show ip route已使用。

N9K1中的路由

```
<#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
```

對於nexus N9K1，使用介面Eth1/1。

N9K2中的路由

```
<#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
```

對於nexus N9K1，使用介面Eth1/1和Eth1/2。

N9K3中的路由

```
<#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
```

對於nexus N9K1，使用介面Eth1/1。

標識資料包大小

若要使用介面計數器排除封包捨棄的疑難問題，需要識別沒有增加的計數器。

在下一個示例中，命令sh interface e1/1 counter detailed運行了兩次，可以觀察到RX和TX的Packets from 512 to 1023 bytes未增加。

此過程需要在源裝置與目的裝置之間的所有相關裝置上完成。

```
<#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

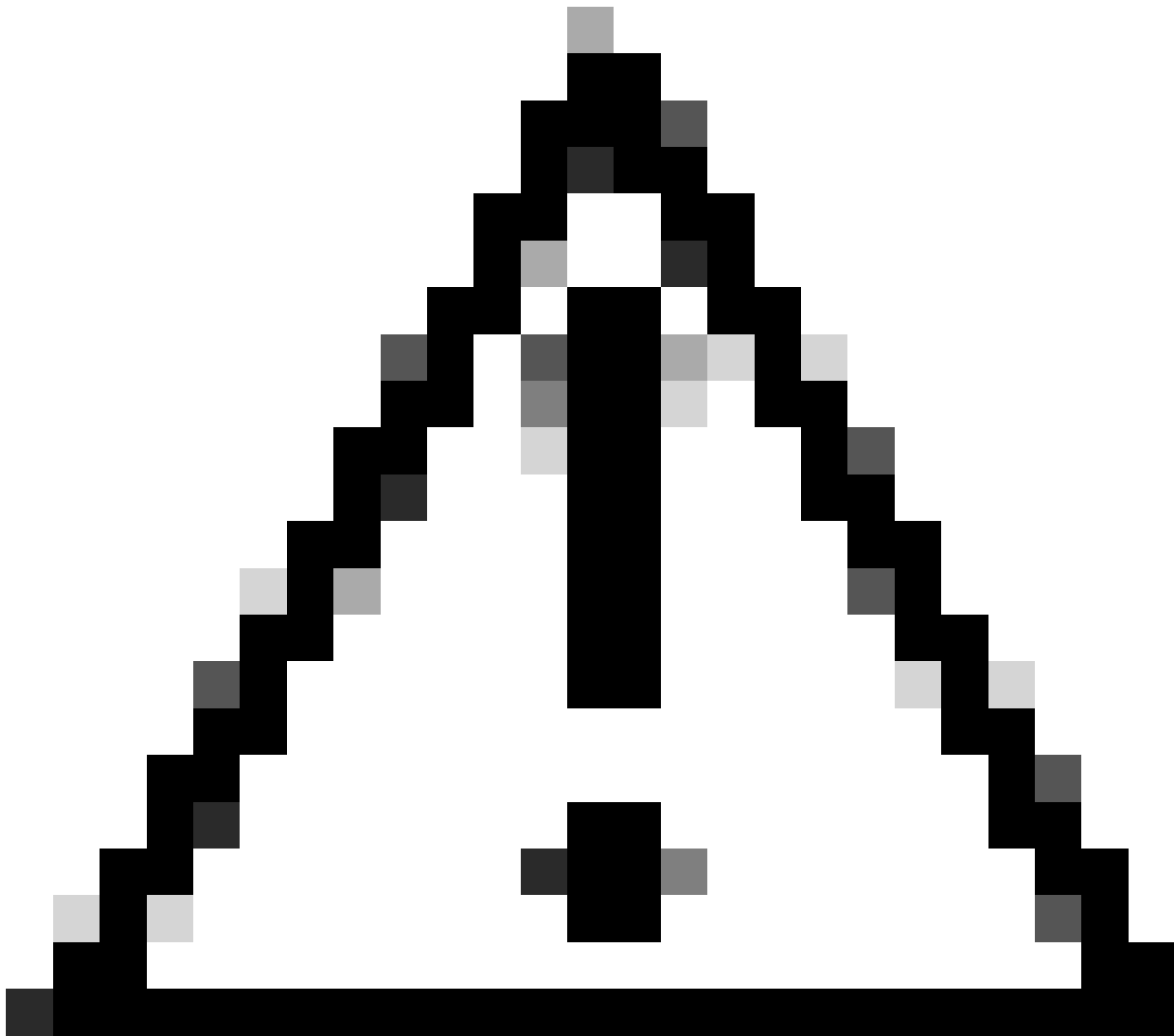
Rx Packets: 31821
Rx Unicast Packets: 8437
Rx Multicast Packets: 23817
Rx Broadcast Packets: 3
Rx Bytes: 8125733
Rx Packets from 0 to 64 bytes: 329
Rx Packets from 65 to 127 bytes: 22878
Rx Packets from 128 to 255 bytes: 3468
Rx Packets from 256 to 511 bytes: 1670

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



注意：在生產環境中，可以清除介面計數器以識別哪個計數器沒有增加。對於將MTU設定為最大值的介面，可以找到大於1518的計數器。如果具有特定大小的資料包沒有通過nexus，則不會出現計數器。

執行測試

對於此測試，由於使用受控環境，因此在所有裝置上都使用計數器Packets from 1024 to 1518。在測試之前清除所有介面的計數器：

```
<#root>
```

```
N9K1
```

```
# clear counters interface
```

```
e1/1
```

```
N9K2
```

```
# clear counters interface
```

```
e1/1-2
```

```
N9K3
```

```
# clear counters interface
```

```
e1/1
```

在所有nexus中，可以運行下一個命令來驗證沒有具有所需資料包大小的流量通過nexus;人們希望看到的不是任何東西；

```
<#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"
```

現在所有計數器都已清除，可以生成ping，指定設定DF-BIT時資料包大小在1024-1518之間。

```
<#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

驗證ICMP請求

在下一個示例中，您可以觀察從N9K1到N9K3的ICMP請求在相關裝置上的計數器如何在TX/RX方向上增加。

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: 5 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>
可以觀察到，N9K1在介面e1/1上傳送了5個資料包	可以觀察到，N9K2在介面e1/1上接收了5個資料包，並在介面e1/2上傳送了5個資料包

驗證ICMP應答

驗證ICMP請求修補程式後，您可以繼續檢視ICMP回覆。

在下一個示例中，您可以觀察從N9K3到N9K1的ICMP應答在相關裝置上的TX/RX方向上計數器如何增加

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>
可以觀察到，N9K1在介面e1/1上接收5個資料包	可以觀察到，N9K2在介面e1/1上傳送了5個資料包，在介面e1/2上接收了5個資料包

通過此測試，可以確認流量是否正確流經3台交換機。如果其中一個nexus在計數器上有差異，則表示流量可能丟棄的RX或TX。

關於此翻譯

思科已使用電腦和人工技術翻譯本文件，讓全世界的使用者能夠以自己的語言理解支援內容。請注意，即使是最佳機器翻譯，也不如專業譯者翻譯的內容準確。Cisco Systems, Inc. 對這些翻譯的準確度概不負責，並建議一律查看原始英文文件（提供連結）。