

Dépannage des pertes de paquets avec les compteurs d'interface sur la plate-forme Nexus

Table des matières

[Introduction](#)

[Conditions préalables](#)

[Exigences](#)

[Composants utilisés](#)

[Topologie](#)

[Fond](#)

[Identification des interfaces](#)

[Routes dans N9K1](#)

[Routes dans N9K2](#)

[Routes dans N9K3](#)

[Identification de la taille des paquets](#)

[Exécution du test](#)

[Vérifier la requête ICMP](#)

[Vérifier la réponse ICMP](#)

Introduction

Ce document décrit comment dépanner la perte de paquets à l'aide des compteurs d'interface Nexus.

Conditions préalables

Exigences

Cisco vous recommande de prendre connaissance des rubriques suivantes :

- Plate-forme NXOS

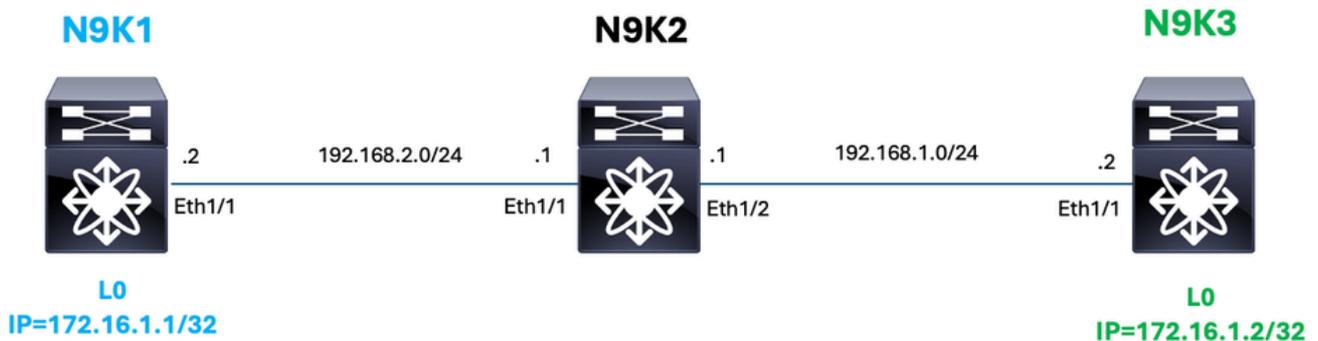
Composants utilisés

| Nom | Plateforme | Version |
|------|-----------------|---------|
| N9K1 | N9K-C93108TC-EX | 9.3(10) |
| N9K2 | N9K-C93108TC-EX | 9.3(10) |
| N9K3 | N9K-C93108TC-EX | 9.3(10) |

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. Si votre réseau

est en ligne, assurez-vous de bien comprendre l'incidence possible des commandes.

Topologie



Fond

Dans certains environnements, les méthodes traditionnelles de capture de paquets telles que ELAM ou SPAN ne peuvent pas être des options réalisables pour diagnostiquer des problèmes réseau. Cependant, les compteurs de paquets de l'interface Nexus constituent une alternative intéressante pour le dépannage des abandons de paquets. Il est important de noter que la disponibilité de compteurs spécifiques peut varier en fonction de la configuration du réseau, de sorte que cette méthode de dépannage ne peut pas être universellement applicable.

Dans cet exemple, il est démontré comment utiliser les compteurs d'interface Nexus pour dépanner des problèmes de connectivité entre les interfaces de bouclage de N9K1 (172.16.1.1) et N9K3 (172.16.1.2).

Identification des interfaces

Pour chaque périphérique, les interfaces d'entrée et de sortie doivent être identifiées, afin d'identifier ces interfaces pour cet exemple de commande : show ip route est utilisé.

Routes dans N9K1

```
<#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
```

Pour le nexus N9K1, l'interface Eth1/1 est utilisée.

Routes dans N9K2

```
<#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
```

Pour le nexus N9K1, les interfaces Eth1/1 et Eth1/2 sont utilisées.

Routes dans N9K3

```
<#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
```

Pour le nexus N9K1, l'interface Eth1/1 est utilisée.

Identification de la taille des paquets

Afin de dépanner les abandons de paquets à l'aide de compteurs d'interface, un compteur qui

n'augmente pas doit être identifié.

Dans l'exemple suivant, la commande `sh interface e1/1 counter detailed` a été exécutée deux fois, où il peut être observé que compteur Packets de 512 à 1023 octets n'a pas augmenté pour RX et TX.

Ce processus doit être effectué sur tous les périphériques impliqués entre la source et la destination.

<#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

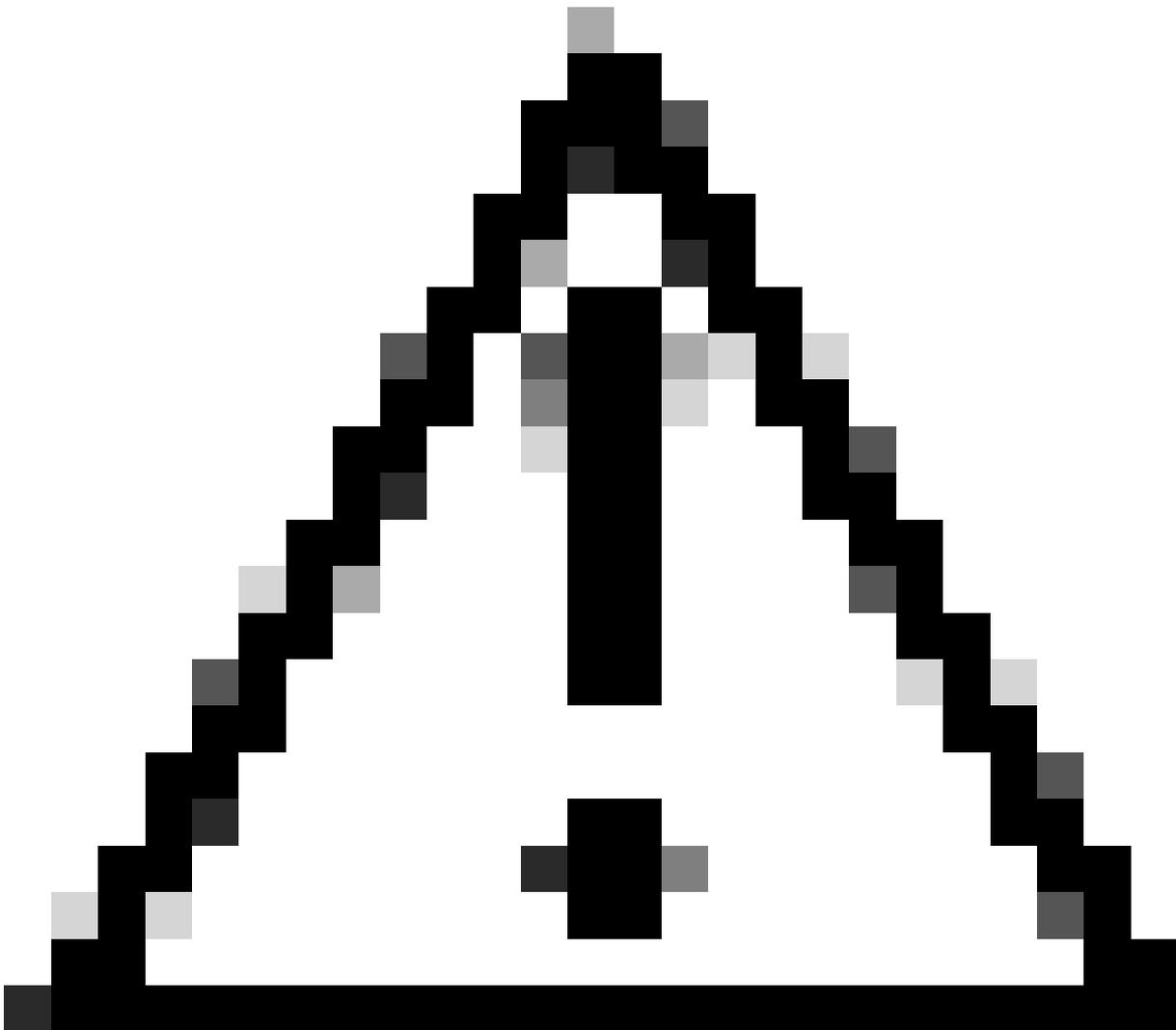
Rx Packets from 512 to 1023 bytes: 63

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



Mise en garde : Dans un environnement de production, les compteurs d'interfaces peuvent être effacés afin d'identifier le compteur qui n'augmente pas. Pour les interfaces dont le MTU est défini sur maximum, des compteurs supérieurs à 1518 peuvent être trouvés. Si des paquets de taille spécifique ne traversent pas le réseau, aucun compteur ne s'affiche.

Exécution du test

Pour ce test, parce qu'un environnement contrôlé est utilisé, compteur Packets from 1024 to 1518 est utilisé dans tous les périphériques. Les compteurs de toutes les interfaces sont effacés avant le test :

```
<#root>
```

```
N9K1
```

```
# clear counters interface
```

```
e1/1
```

```
N9K2
```

```
# clear counters interface
```

```
e1/1-2
```

```
N9K3
```

```
# clear counters interface
```

```
e1/1
```

Dans tous les nexus, la commande suivante peut être exécutée pour vérifier qu'aucun trafic avec la taille de paquet souhaitée ne passe par le nexus ; l'attente est de ne rien voir ;

```
<#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"
```

Maintenant que tous les compteurs sont libres, une requête ping peut être générée , spécifiant une taille de paquet comprise entre 1024-1518 avec DF-BIT défini.

<#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

Vérifier la requête ICMP

Dans l'exemple suivant, vous pouvez observer comment les compteurs augmentent dans la direction TX/RX sur les périphériques impliqués pour la requête ICMP de N9K1 à N9K3.

| N9K1 | N9K2 |
|---|---|
| <pre><#root> N9K1# sh int e1/1 cou detailed i i " 1024 to 1518" 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" 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" Rx Packets from 1024 to 1518 bytes: 0 Tx Packets from 1024 to 1518 bytes:</pre> |

| | |
|--|---|
| | 5 |
| On constate que N9K1 a envoyé 5 paquets sur l'interface e1/1 | On observe que N9K2 a reçu 5 paquets sur l'interface e1/1 et en a envoyé 5 sur l'interface e1/2 |

Vérifier la réponse ICMP

Une fois le correctif de la demande ICMP validé, vous pouvez passer en revue la réponse ICMP.

Dans l'exemple suivant, vous pouvez observer comment les compteurs augmentent dans la direction TX/RX sur les périphériques impliqués pour la réponse ICMP de N9K3 à N9K1

| 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> |
| On observe que N9K1 reçoit 5 paquets sur l'interface e1/1 | On peut observer que N9K2 a envoyé 5 paquets sur l'interface e1/1 et a reçu 5 paquets sur l'interface e1/2 |

Avec ce test, il est possible de confirmer que le trafic a circulé correctement sur les 3 commutateurs. Si l'un des nexus a une divergence sur le compteur, RX ou TX est l'endroit où le trafic peut être abandonné.

À propos de cette traduction

Cisco a traduit ce document en traduction automatisée vérifiée par une personne dans le cadre d'un service mondial permettant à nos utilisateurs d'obtenir le contenu d'assistance dans leur propre langue.

Il convient cependant de noter que même la meilleure traduction automatisée ne sera pas aussi précise que celle fournie par un traducteur professionnel.