

Dépannez le paquet Ethernet corrompu sur Cisco Nexus 9000

Contenu

[Introduction](#)

[Informations générales](#)

[Va comment un paquet traité par un commutateur](#)

[Remplissage modifiée avec des VLAN étiquetés quand le trafic traverse N9K](#)

[Solution](#)

Introduction

Ce document décrit comment dépanner le paquet Ethernet corrompu sur Cisco Nexus 9000 quand des informations de remplissage sont corrompues ou mal formées.

[Informations générales](#)

La taille minimale d'une trame Ethernet est de 64 octets, aucune balise de la matière VLAN n'est présente là ou pas.

La taille minimale de charge utile d'Ethernets est :

- 46 octets si la balise VLAN est absente.
- 42 octets si la balise VLAN est présente.

Vous pouvez vérifier ce fait :

- Sur Wikipedia, **charge utile de** section : https://en.wikipedia.org/wiki/Ethernet_frame
- Sur la norme d'IEEE 802.3 (http://people.ee.duke.edu/~mbrooke/EE164.02/Spring_2004/group_2/index_files/8023.pdf), où le format de trame MAC (sans VLAN) est défini dans la section 3.1.1, la page 39, et les éléments d'une trame MAC étiquetée est défini à la section 3.5 de la page 43.

La taille minimale d'un paquet Ethernet est de 64 octets, aucune en-tête de la matière VLAN n'est présente là ou pas. On permet au le serveur pour envoyer un paquet de 64 octets de long qui contient un VLAN, que vous devriez recevoir et traiter correctement.

Note: Ce comportement est correctement manipulé par un Catalyst 4500x pas par le Nexus 9k.

Va comment un paquet traité par un commutateur

Étape 1. Recevez une trame Ethernet **VALIDE** de 64 octets.

Étape 2. Retirez le Frame Check Sequence (FCS), ainsi le paquet devient de 60 octets de long.

Étape 3. Retirez la balise VLAN, ainsi le paquet devient de 56 octets de long.

Étape 4. Ajoutez la remplissage pour rendre le paquet 60 octets long.

Étape 5. Il ajoute la FCS, rendant le paquet 64 octets long.

La remplissage ne devrait pas obtenir modifié quand un paquet passe par le commutateur cut-through.

Remplissage modifiée avec des VLAN étiquetés quand le trafic traverse N9K

Au lieu de la remplissage avec des zéros, le paquet est complété avec des caractères inutilisables, dans la plupart des cas il n'a aucune incidence parce que des sommes de contrôle ne sont pas modifiées et ainsi personne n'utilise ces données. Cependant, si les clients ont une utilisation spéciale et doivent recomputer les sommes de contrôle, ces données de déchets mènent à la corruption des sommes de contrôle à la fin (d'autres appliances, comme NAT/load-balancers pourraient voir la question trop)

Le périphérique est un N9K 93120TX (a été au commencement détecté sur un 9372TX cependant), version est le dernier NXOS 7.0(3)I2(2a)

Hôtes de Linux d'utilisation avec le matériel ici directement connecté au N9K (aucune virtualisation de sorte) (liens 1000base-T)

Utilisez cette configuration :

```
interface Ethernet1/59
    switchport mode trunk
!
interface Ethernet1/60
    switchport mode trunk
```

linux configurations:

```
inet 10.2.1.1/24 brd 10.2.1.255 scope global eth1 <= native vlan
inet 10.1.1.1/24 brd 10.1.1.255 scope global eth1.100 <= tagged vlan 100
```

ou

Juste connectez l'hôte de fenêtres et envoyez les trames marquées, elles devrait déclencher le problème. D'ailleurs, assurez-vous que le network interface card (NIC) a la capacité d'étiqueter le paquet.

Le commutateur ajoute la remplissage différente de zéro aux trames qui traverse.

Exemple : Hôte — — [joncteur réseau] N9K [joncteur réseau] — — hôte

Vous pouvez employer le netcat pour envoyer et recevoir les paquets.

Suivant les indications de l'image, il envoie le côté (VLAN 100 étiqueté), mettent en communication e1/59 sur le commutateur

```
6: eth1.100@eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9000 qdisc noqueue state UP group default
link/ether 44:a8:42:2c:5f:c4 brd ff:ff:ff:ff:ff:ff
inet 10.1.1.1/24 brd 10.1.1.255 scope global eth1.100
    valid_lft forever preferred_lft forever
inet6 fe80::46a8:42ff:fe2c:5fc4/64 scope link
    valid_lft forever preferred_lft forever
```

```
root@s35-c2-0:~# nc 10.1.1.2 3002 -u
a
^C
root@s35-c2-0:~#
```

Il reçoit le côté (VLAN 100 étiqueté), le port e1/60 sur le commutateur, suivant les indications de l'image :

```
7: eth1.100@eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9000 qdisc noqueue state UP group default
link/ether 44:a8:42:2c:63:d1 brd ff:ff:ff:ff:ff:ff
inet 10.1.1.2/24 brd 10.1.1.255 scope global eth1.100
    valid_lft forever preferred_lft forever
inet6 fe80::46a8:42ff:fe2c:63d1/64 scope link
    valid_lft forever preferred_lft forever
```

```
root@s35-c2:~# nc -l -u -p 3002
a
^C
root@s35-c2:~#
```

Suivant les indications de l'image, le paquet est transmis.

```
root@s35-c2-0:~# tcpdump -i eth1.100 -nvex
tcpdump: listening on eth1.100, link-type EN10MB (Ethernet), capture size 65535 bytes
10:42:20.953994 44:a8:42:2c:5f:c4 > 44:a8:42:2c:63:d1, ethertype IPv4 (0x0800), length 44: (tos 0x0, ttl 64, id 64283, offset 0, flags [DF], proto UDP (17), length 30)
    10.1.1.1.41675 > 10.1.1.2.3002: UDP, length 2
    0x0000: 4500 001e fb1b 4000 4011 29af 0a01 0101
    0x0010: 0a01 0102 a2cb 0bba 000a 1620 610a
^C
1 packet captured
1 packet received by filter
0 packets dropped by kernel
root@s35-c2-0:~#
```

Le paquet est reçu, suivant les indications de l'image :

```

10:43:12.665897 44:a8:42:2c:5f:c4 > 44:a8:42:2c:63:d1, ethertype IPv4 (0x0800), length 60: (tos 0x0, ttl 64, id 64283, offset 0, flags [DF], proto UDP
(17), length 30)
  10.1.1.1.41675 > 10.1.1.2.3002: UDP, length 2
    0x0000: 4500 001e fb1b 4000 4011 29af 0a01 0101
    0x0010: 0a01 0102 a2cb 0bba 000a da45 610a 0000
    0x0020: 0000 0000 0000 0000 0000 7562 710e
^C
7 packets captured
7 packets received by filter
0 packets dropped by kernel
root@s35-c2:~#

```

Suivant les indications de l'image, la remplissage fausse est mise en valeur.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	10.1.1.1	10.1.1.2	UDP	60	Source port: 40849 Destination port: 3002

```

> Frame 1: 60 bytes on wire (480 bits), 60 bytes captured (480 bits)
v Ethernet II, Src: Dell_2c:5f:c4 (44:a8:42:2c:5f:c4), Dst: Dell_2c:63:d1 (44:a8:42:2c:63:d1)
  > Destination: Dell_2c:63:d1 (44:a8:42:2c:63:d1)
  > Source: Dell_2c:5f:c4 (44:a8:42:2c:5f:c4)
  Type: IP (0x0800)
  Padding: 0000000000000000000000000000f1b7bc5c
v Internet Protocol Version 4, Src: 10.1.1.1 (10.1.1.1), Dst: 10.1.1.2 (10.1.1.2)
  0100 ... = Version: 4
  ... 0101 = Header Length: 20 bytes
  > Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00: Not-ECT (Not ECN-Capable Transport))
  Total Length: 30
  Identification: 0xfb1d (64285)
  > Flags: 0x02 (Don't Fragment)
  Fragment offset: 0
  Time to live: 64
  Protocol: UDP (17)
  > Header checksum: 0x29ad [validation disabled]
  Source: 10.1.1.1 (10.1.1.1)
  Destination: 10.1.1.2 (10.1.1.2)
  [Source GeoIP: Unknown]
  [Destination GeoIP: Unknown]
v User Datagram Protocol, Src Port: 40849 (40849), Dst Port: 3002 (3002)
  Source Port: 40849 (40849)
  Destination Port: 3002 (3002)
  Length: 10
  > Checksum: 0xdd7f [validation disabled]
  [Good Checksum: False]
  [Bad Checksum: False]
  [Stream index: 0]
v Data (2 bytes)
  Data: 610a
  [Length: 2]
0000 44 a8 42 2c 63 d1 44 a8 42 2c 5f c4 00 00 45 00  D..c.D. 0,.....E.
0010 00 1e fb 1d 40 00 40 11 29 ad 0a 01 01 01 0a 01  ....@.@. ].....
0020 01 02 9f 91 0b ba 00 0a dd 7f 61 0a 00 00 00 00  .....0.....
0030 00 00 00 00 00 00 00 00 f1 b7 bc 5c  .......X

```

Ceci est également affiché avec un analyseur de paquet (en un autre paquet, les données sont différentes des captures d'écran précédentes mais le test et la bogue est identique),

Solution

Le travail est autour de désactiver la mémoire-poussée sur l'interface où nous avons ceci serveur connecté.

C9396PX-1(config)# international et 1/7

C9396PX-1(config-if)# aucune mémoire-poussée

Défaut relatif :

[CSCva46849](#) vue de 60 octets avec la commutation de l'en-tête L2 dot1q sur N9k