

# Risoluzione dei problemi relativi al pacchetto Ethernet danneggiato su Cisco Nexus 9000

## Sommario

[Introduzione](#)

[Premesse](#)

[Come viene elaborato un pacchetto da uno switch](#)

[Spaziatura interna modificata con VLAN con tag quando il traffico attraversa N9K](#)

[Soluzione](#)

## Introduzione

In questo documento viene descritto come risolvere i problemi relativi al pacchetto ethernet danneggiato in Cisco Nexus 9000 quando le informazioni di spaziatura interna sono danneggiate o in formato non corretto.

## Premesse

La dimensione minima di un frame Ethernet è di 64 byte, a prescindere dal tag VLAN presente o meno.

Le dimensioni minime del payload Ethernet sono:

- 46 byte se il tag VLAN è assente.
- 42 byte se il tag VLAN è presente.

È possibile verificare questo fatto:

- Su Wikipedia, la sezione **Payload**: [https://en.wikipedia.org/wiki/Ethernet\\_frame](https://en.wikipedia.org/wiki/Ethernet_frame)
- Nello standard IEEE 802.3 ([http://people.ee.duke.edu/~mbrooke/EE164.02/Spring\\_2004/group\\_2/index\\_files/8023.pdf](http://people.ee.duke.edu/~mbrooke/EE164.02/Spring_2004/group_2/index_files/8023.pdf)), dove il formato del frame MAC (senza VLAN) è definito nella sezione 3.1.1, pagina 39, e gli elementi di un frame MAC con tag sono definiti a pagina 43, sezione 3.5.

Le dimensioni minime di un pacchetto Ethernet sono di 64 byte, indipendentemente dalla presenza dell'intestazione VLAN. Il server può inviare un pacchetto da 64 byte contenente una VLAN che deve essere accettato ed elaborato correttamente.

**Nota:** Questo comportamento viene gestito correttamente da un Catalyst 4500x e non da Nexus 9k.

## Come viene elaborato un pacchetto da uno switch

Passaggio 1. Ricevere un frame Ethernet da 64 byte **VALIDO**.

Passaggio 2. Rimuovere la sequenza Frame Check (FCS), in modo che il pacchetto diventi lungo 60 byte.

Passaggio 3. Rimuovere il tag VLAN, in modo che il pacchetto diventi lungo 56 byte.

Passaggio 4. Aggiungere il padding per rendere il pacchetto lungo 60 byte.

Passaggio 5. Aggiunge il file FCS, rendendo il pacchetto lungo 64 byte.

Il padding non deve essere modificato quando un pacchetto passa attraverso lo switch cut-through.

## Spaziatura interna modificata con VLAN con tag quando il traffico attraversa N9K

Invece di riempire con zeri, il pacchetto viene riempito con caratteri garbage, nella maggior parte dei casi non ha alcun impatto perché i checksum non vengono modificati e quindi nessuno usa questi dati. Tuttavia, se i clienti hanno un utilizzo speciale e devono ricalcolare i checksum, questi dati di garbage portano al danneggiamento dei checksum alla fine (altri accessori, come NAT/load-balancers potrebbero vedere il problema anche).

Il dispositivo è un N9K 93120TX (inizialmente rilevato su un 9372TX sebbene), la versione è la più recente NXOS 7.0(3)I2(2a).

Utilizzare host Linux con hardware connesso direttamente a N9K (nessuna virtualizzazione di alcun tipo) qui (collegamenti 1000base-T).

Utilizzare questa configurazione:

```
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
```

o

È sufficiente connettere l'host Windows e inviare i frame con tag, in modo che il problema venga attivato. Verificare inoltre che la scheda di interfaccia di rete (NIC) sia in grado di etichettare il pacchetto.

L'opzione Switch aggiunge la spaziatura interna diversa da zero ai fotogrammi attraversati.

Ad esempio: Host — [Trunk] N9K [Trunk] — Host

È possibile utilizzare netcat per inviare e ricevere i pacchetti.

Come mostrato nell'immagine, invia un segnale laterale (VLAN 100 con tag), porta e1/59 sullo switch.

```
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:~#
```

Riceve una porta laterale (VLAN 100 con tag), la porta e1/60 sullo switch, come mostrato nell'immagine:

```
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:~#
```

Come mostrato nell'immagine, il pacchetto viene trasmesso.

```
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:~#
```

Il pacchetto viene ricevuto, come mostrato nell'immagine:

```

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:~#

```

Come mostrato nell'immagine, viene evidenziata la spaziatura interna errata.

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)
▼ 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: 000000000000000000000000f1b7bc5c
▼ 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]
▼ 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]
▼ Data (2 bytes)
  Data: 610a
  [Length: 2]

```

---

```

0000 44 a8 42 2c 63 d1 44 a8 42 2c 5f c4 08 00 45 00  D.B,c.D. B,....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  .....a.....
0030 00 00 00 00 00 00 00 00 f1 b7 bc 5c          ..... \

```

Questa condizione viene visualizzata anche con un analizzatore di pacchetti (in un altro pacchetto i dati sono diversi dalle schermate precedenti, ma il test e il bug sono identici),

## Soluzione

La soluzione è disabilitare l'[aumento del buffer](#) sull'interfaccia a cui è connesso il server.

```

C9396PX-1(config)# int et 1/7
C9396PX-1(config-if)# no buffer-boost

```

**Difetto correlato:**

[CSCva46849](#) 60 Byte Frame con dot1q Header L2 attivato su N9k