Risoluzione dei problemi di lentezza TCP dovuti alla regolazione del valore MSS sugli switch Catalyst 9K

Sommario

Introduzione

In questo documento viene descritto come uno switch Catalyst 9K esegue la regolazione del valore TCP MSS e come la lentezza TCP viene collegata a questa funzione.

Informazioni sulla regolazione TCP MSS

La funzione di regolazione Maximum Segment Size (MSS) del protocollo TCP (Transmission Control Protocol) consente di configurare le dimensioni massime del segmento per i pacchetti temporanei che attraversano un router, in particolare i segmenti TCP con bit SYN impostato. Il comando`ip tcp adjust-mss`viene usato in modalità di configurazione interfaccia per specificare il valore MSS sul router intermedio dei pacchetti SYN in modo da evitare il troncamento.

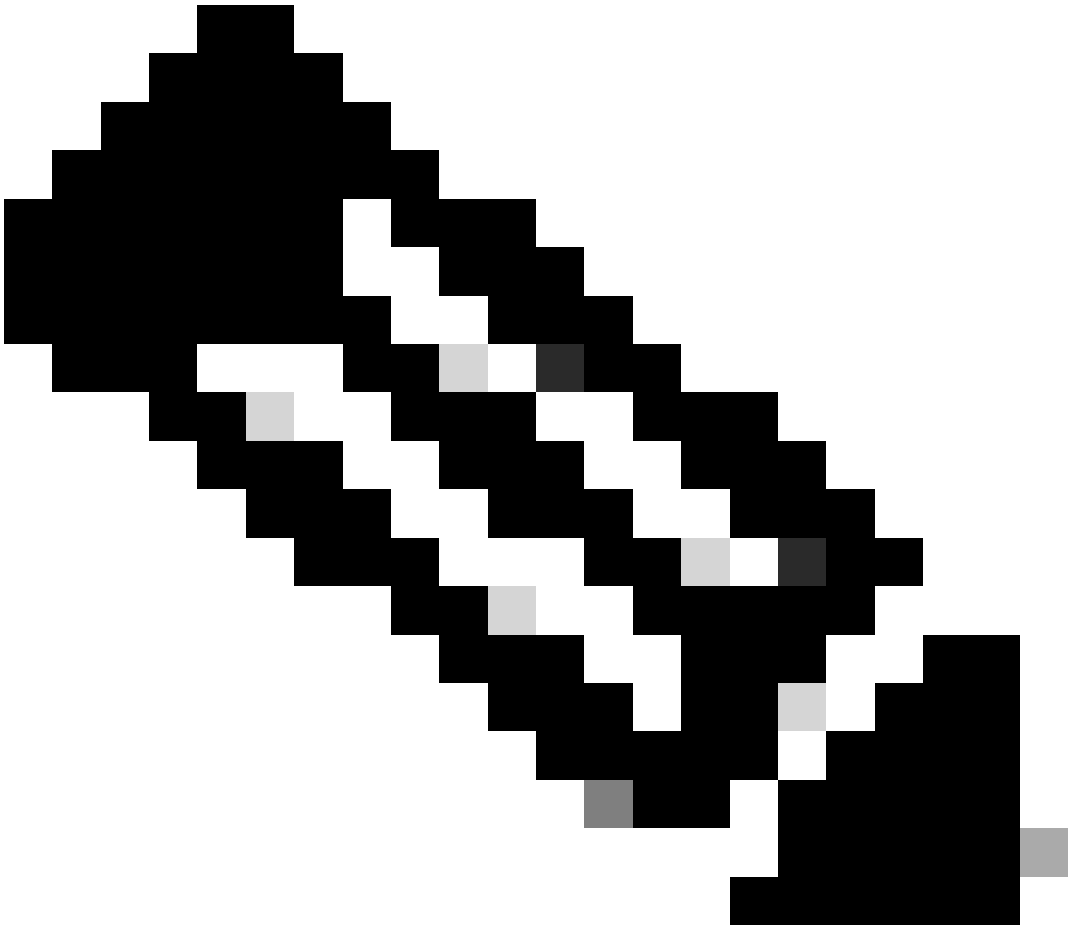
Quando un host (in genere un PC) avvia una sessione TCP con un server, negozia le dimensioni del segmento IP utilizzando il campo dell'opzione MSS nel pacchetto TCP SYN. La configurazione MTU sull'host determina il valore del campo MSS. Il valore MTU predefinito per una scheda NIC del PC è 1500 byte con un valore TCP MSS di 1460 (1500 byte - 20 byte intestazione IP - 20 byte intestazione TCP).

Lo standard PPPoE (PPP over Ethernet) supporta una MTU di soli 1492 byte.

La disparità tra le dimensioni dell'MTU PPPoE e dell'host può causare il rifiuto da parte del router dei pacchetti da 1500 byte tra l'host e il server e l'interruzione delle sessioni TCP sulla rete PPPoE.

Anche se sull'host è abilitata l'MTU del percorso (che rileva la MTU corretta sul percorso), le sessioni possono essere eliminate perché a volte gli amministratori di sistema disabilitano i messaggi di errore ICMP (Internet Control Message Protocol) che devono essere inoltrati dall'host perché la MTU del percorso funzioni.

Il comando ip tcp adjust-mss aiuta a evitare che le sessioni TCP vengano interrotte regolando il valore MSS dei pacchetti TCP SYN. il comando ip tcp adjust-mss ha effetto solo sulle connessioni TCP che passano attraverso il router. Nella maggior parte dei casi, il valore ottimale per l'argomento max-segment-size del comando ip tcp adjust-mss è 1452 byte.
Questo valore, sommato all'intestazione IP da 20 byte, all'intestazione TCP da 20 byte e all'intestazione PPPoE da 8 byte, aggiunge un pacchetto da 1500 byte che corrisponde alle dimensioni MTU del collegamento Ethernet.



Nota: sugli switch Catalyst 9K il traffico basato sulla regolazione TCP MSS è commutato dal software. In questo documento vengono illustrati alcuni scenari in cui si presume che il traffico basato sulle regolazioni TCP MSS sia commutato dal software. Per verificare se un software HW/SW specifico commuta il traffico basato sulla regolazione TCP MSS, consultare la guida alla configurazione.

Comportamento

Come accennato in precedenza, il traffico basato sulla regolazione TCP MSS è sempre commutato dal software.
Ciò significa che se si tenta di eseguire la regolazione TCP, lo switch invia il traffico TCP alla CPU per la modifica del valore MSS.

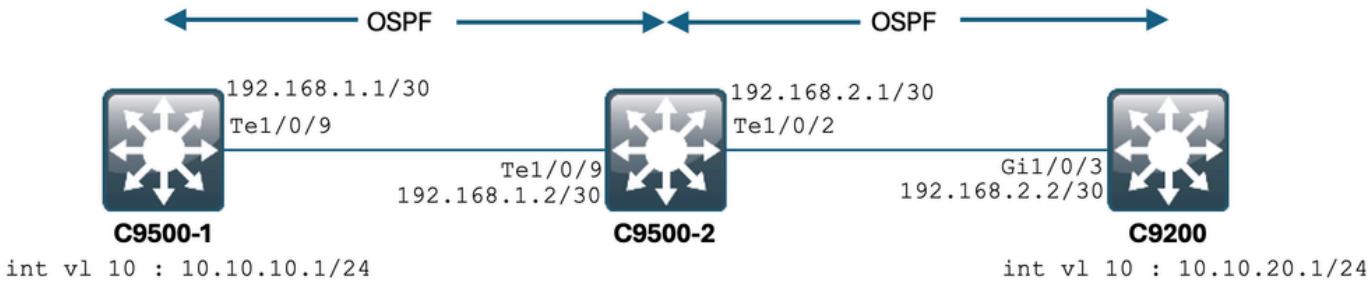
Ad esempio, se si modifica il valore TCP MSS su un'interfaccia, tutto il traffico TCP ricevuto su quell'interfaccia viene puntato alla CPU.
La CPU quindi modifica il valore MSS e invia il traffico all'interfaccia richiesta a cui era diretto il pacchetto TCP.

Per questo motivo, se c'è una quantità enorme di traffico TCP con regolazione MSS, allora questo sovraccarica la coda della CPU.
Quando una coda CPU è sovraccarica, il controller COPP (Control Plane Policer) esegue il traffico e scarta i pacchetti per mantenere la velocità del controller di coda. In questo modo, i pacchetti TCP vengono scartati.

Di conseguenza, vengono rilevati problemi quali la lentezza del trasferimento dei file, la creazione di sessioni SSH e la lentezza delle applicazioni Citrix (se si utilizza TCP).

Di seguito è riportato un esempio reale di come ciò accada.

Topologia



Scenario

Si sta per passare al protocollo SSH nel C9200 dal C9500-1.
SSH utilizzando la VLAN 10 (10.10.10.1) del C9500-1 come origine.
La destinazione del protocollo SSH è la VLAN 20 del C9200 (10.10.20.1/24).
SSH è basato su TCP, quindi qualsiasi lentezza nel protocollo TCP influisce anche sulla creazione della sessione SSH.

È presente uno switch L3 in transito (C9500-2) tra C9500-1 e C9200.
Esistono due collegamenti di transito/30 L3, uno tra C9500-1 e C9500-2 e uno tra C9500-2 e

C9200.
Il protocollo OSPF viene utilizzato per garantire la raggiungibilità su tutti e tre gli switch. Tutte le subnet IP/30 e le SVI vengono pubblicizzate nel protocollo OSPF.
Tutti gli IP mostrati in precedenza sono raggiungibili tra loro.

In C9500-2 Te1/0/9, viene modificato il valore TCP MSS.
Quando si avvia SSH da C9500-1, si verifica un handshake TCP a 3 vie.
Il pacchetto SYN raggiunge il C9500-2 Te1/0/9 (Ingress), dove viene eseguita la regolazione del parametro TCP MSS.

Configurazione iniziale e comportamento

È stata eseguita un'acquisizione EPC su C9500-2 Te1/0/9 (entrambe le direzioni) ed è stato avviato SSH da C9500-1 a C9200.

Di seguito è riportata la configurazione EPC:

```
C9500-2#show monitor capture mycap
Status Information for Capture mycap
Target Type:
Interface: TenGigabitEthernet1/0/9, Direction: BOTH
Status : Inactive
Filter Details:
Capture all packets
Buffer Details:
Buffer Type: LINEAR (default)
Buffer Size (in MB): 80
File Details:
File not associated
Limit Details:
Number of Packets to capture: 0 (no limit)
Packet Capture duration: 0 (no limit)
Packet Size to capture: 0 (no limit)
Maximum number of packets to capture per second: 1000
Packet sampling rate: 0 (no sampling)
C9500-2#
```

Avvio dell'EPC:

```
C9500-2#monitor capture mycap start
Started capture point : mycap
C9500-2#
```

Avvio del protocollo SSH da C9500-1 a C9200:

```
C9500-1#ssh -l admin 10.10.20.1
```

```
Password:
```

Arresto dell'EPC:

```
C9500-2#monitor capture mycap stop
Capture statistics collected at software:
Capture duration - 6 seconds
Packets received - 47
Packets dropped - 0
Packets oversized - 0
Bytes dropped in asic - 0
Capture buffer will exists till exported or cleared
Stopped capture point : mycap
C9500-2#
```

Ecco i pacchetti acquisiti da EPC:

```
C9500-2#show monitor capture mycap buffer brief
Starting the packet display ........ Press Ctrl + Shift + 6 to exit
1 0.000000 10.10.10.1 -> 10.10.20.1 TCP 60 44274 -> 22 [SYN] Seq=0 Win=4128 Len=0 MSS=536
2 0.001307 10.10.20.1 -> 10.10.10.1 TCP 60 22 -> 44274 [SYN, ACK] Seq=0 Ack=1 Win=4128 Len=0 MSS=536
3 0.001564 10.10.10.1 -> 10.10.20.1 TCP 60 44274 -> 22 [ACK] Seq=1 Ack=1 Win=4128 Len=0
4 0.003099 10.10.20.1 -> 10.10.10.1 SSH 73 Server: Protocol (SSH-2.0-Cisco-1.25)
5 0.003341 10.10.10.1 -> 10.10.20.1 SSH 73 Client: Protocol (SSH-2.0-Cisco-1.25)
6 0.003419 10.10.10.1 -> 10.10.20.1 TCP 118 [TCP segment of a reassembled PDU]
7 0.003465 10.10.10.1 -> 10.10.20.1 TCP 118 44274 -> 22 [ACK] Seq=84 Ack=20 Win=4109 Len=64 [TCP segmen
8 0.003482 10.10.10.1 -> 10.10.20.1 TCP 118 44274 -> 22 [ACK] Seq=148 Ack=20 Win=4109 Len=64 [TCP segme
9 0.003496 10.10.10.1 -> 10.10.20.1 TCP 118 44274 -> 22 [ACK] Seq=212 Ack=20 Win=4109 Len=64 [TCP segme
10 0.003510 10.10.10.1 -> 10.10.20.1 TCP 118 44274 -> 22 [ACK] Seq=276 Ack=20 Win=4109 Len=64 [TCP segm
11 0.003525 10.10.10.1 -> 10.10.20.1 TCP 118 44274 -> 22 [ACK] Seq=340 Ack=20 Win=4109 Len=64 [TCP segm
12 0.004719 10.10.20.1 -> 10.10.10.1 TCP 60 22 -> 44274 [ACK] Seq=20 Ack=84 Win=4045 Len=0
~ Output Cut ~
```

Potete vedere l'handshake TCP che avviene nel pacchetto numero 1,2,3.
Il pacchetto SYN è il pacchetto N. 1.
Come si può vedere, il valore MSS è 536.
Il pacchetto SYN, ACK (Pacchetto n. 2) proviene anche dal C9200 con un valore MSS di 536.
In questo caso, il valore MSS rimane invariato e non viene modificato dallo switch.

Comportamento dopo la regolazione TCP MSS

Ecco la configurazione della regolazione TCP MSS su C9500-2 Te1/0/9:

```
C9500-2#sh run int te1/0/9
Building configuration...
Current configuration : 119 bytes
```

```
!
interface TenGigabitEthernet1/0/9
no switchport
ip address 192.168.1.2 255.255.255.252
ip tcp adjust-mss 512 ------------------> Here we are changing the MSS value to 512.
```

A questo punto, eseguire un'acquisizione EPC su C9500-2 Te1/0/9 (in entrambe le direzioni) e avviare SSH da C9500-1 a C9200.

Di seguito è riportata la configurazione EPC:

```
C9500-2#show monitor capture mycap
Status Information for Capture mycap
Target Type:
Interface: TenGigabitEthernet1/0/9, Direction: BOTH
Status : Inactive
Filter Details:
Capture all packets
Buffer Details:
Buffer Type: LINEAR (default)
Buffer Size (in MB): 80
File Details:
File not associated
Limit Details:
Number of Packets to capture: 0 (no limit)
Packet Capture duration: 0 (no limit)
Packet Size to capture: 0 (no limit)
Maximum number of packets to capture per second: 1000
Packet sampling rate: 0 (no sampling)
C9500-2#
```

Avviare l'acquisizione, SSH da C9500-1 a C9200 e arrestare l'acquisizione.

Di seguito sono riportati i pacchetti acquisiti dalla CPU:

```
C9500-2#show monitor capture mycap buffer brief
Starting the packet display ........ Press Ctrl + Shift + 6 to exit
1 0.000000 b8:a3:77:ec:ba:f7 -> 01:00:0c:cc:cc:cc CDP 398 Device ID: C9500-1.cisco.com Port ID: TenGiga
2 0.636138 10.10.10.1 -> 10.10.20.1 TCP 60 53865 -> 22 [SYN] Seq=0 Win=4128 Len=0 MSS=536
3 0.637980 10.10.20.1 -> 10.10.10.1 TCP 60 22 -> 53865 [SYN, ACK] Seq=0 Ack=1 Win=4128 Len=0 MSS=512
4 0.638214 10.10.10.1 -> 10.10.20.1 TCP 60 53865 -> 22 [ACK] Seq=1 Ack=1 Win=4128 Len=0
5 0.639997 10.10.20.1 -> 10.10.10.1 SSH 73 Server: Protocol (SSH-2.0-Cisco-1.25)
6 0.640208 10.10.10.1 -> 10.10.20.1 SSH 73 Client: Protocol (SSH-2.0-Cisco-1.25)
7 0.640286 10.10.10.1 -> 10.10.20.1 TCP 118 [TCP segment of a reassembled PDU]
8 0.640341 10.10.10.1 -> 10.10.20.1 TCP 118 53865 -> 22 [ACK] Seq=84 Ack=20 Win=4109 Len=64 [TCP segmen
9 0.640360 10.10.10.1 -> 10.10.20.1 TCP 118 53865 -> 22 [ACK] Seq=148 Ack=20 Win=4109 Len=64 [TCP segme
10 0.640375 10.10.10.1 -> 10.10.20.1 TCP 118 53865 -> 22 [ACK] Seq=212 Ack=20 Win=4109 Len=64 [TCP segm
11 0.640390 10.10.10.1 -> 10.10.20.1 TCP 118 53865 -> 22 [ACK] Seq=276 Ack=20 Win=4109 Len=64 [TCP segm
12 0.640410 10.10.10.1 -> 10.10.20.1 TCP 118 53865 -> 22 [ACK] Seq=340 Ack=20 Win=4109 Len=64 [TCP segm
~ Output Cut ~
```

Potete vedere l'handshake TCP che avviene nei pacchetti numero 2,3,4.
Il pacchetto n. 2 è il pacchetto SYN.
Come si può vedere, il valore MSS è 536.
Tuttavia, il pacchetto SYN, ACK (pacchetto n. 3) proviene dal C9200 con un valore MSS di 512.
Infatti, quando il pacchetto SYN raggiunge il C9500-2 Te1/0/9, viene inviato alla CPU del C9500-2 per la modifica del valore TCP MSS da 536 a 512.
La CPU del C9500-2 cambia il valore MSS a 512 e invia il pacchetto SYN da Te1/0/2 a C9200.
Quindi tutte le transazioni TCP successive usano lo stesso valore MSS modificato.

A questo punto, è possibile esaminare in dettaglio il modo in cui il pacchetto SYN attraversa lo switch e la modifica del valore MSS si verifica.
Quando il pacchetto SYN raggiunge l'interfaccia del C9500-2, viene inviato alla CPU per la modifica del valore MSS.
Prima passa attraverso il FED (dove è possibile catturarlo), quindi passa alla CPU (dove è possibile catturarlo).

Prendiamo prima una cattura di FED Punt su C9500-2.

Di seguito è riportata la configurazione di FED Punt Capture:

```
C9500-2#debug platform software fed switch 1 punt packet-capture buffer limit 16384
Punt PCAP buffer configure: one-time with buffer size 16384...done
```

Avvio della cattura del punt da parte della FED:

```
C9500-2#debug platform software fed switch 1 punt packet-capture start
Punt packet capturing started.
```

Avvio del protocollo SSH da C9500-1 a C9200:

```
C9500-1#ssh -l admin 10.10.20.1
Password:
```

Interruzione della cattura del punt della FED:

```
C9500-2#debug platform software fed switch 1 punt packet-capture stop
Punt packet capturing stopped. Captured 3 packet(s)
```

E questi sono i pacchetti acquisiti dalla FED:

```
C9500-2#show platform software fed switch active punt packet-capture brief
Punt packet capturing: disabled. Buffer wrapping: disabled
Total captured so far: 3 packets. Capture capacity : 16384 packets

------ Punt Packet Number: 1, Timestamp: 2024/07/31 01:29:46.373 ------
interface : physical: TenGigabitEthernet1/0/9[if-id: 0x00000040], pal: TenGigabitEthernet1/0/9 [if-id: 
metadata : cause: 55 [For-us control], sub-cause: 0, q-no: 4, linktype: MCP_LINK_TYPE_IP [1]
ether hdr : dest mac: 0100.5e00.0005, src mac: b8a3.77ec.baf7
ether hdr : ethertype: 0x0800 (IPv4)
ipv4 hdr : dest ip: 224.0.0.5, src ip: 192.168.1.1
ipv4 hdr : packet len: 100, ttl: 1, protocol: 89

------ Punt Packet Number: 2, Timestamp: 2024/07/31 01:29:47.432 ------
interface : physical: TenGigabitEthernet1/0/9[if-id: 0x00000040], pal: TenGigabitEthernet1/0/9 [if-id: 
metadata : cause: 11 [For-us data], sub-cause: 1, q-no: 14, linktype: MCP_LINK_TYPE_IP [1]
ether hdr : dest mac: 00a3.d144.4bf7, src mac: b8a3.77ec.baf7
ether hdr : ethertype: 0x0800 (IPv4)
ipv4 hdr : dest ip: 10.10.20.1, src ip: 10.10.10.1
ipv4 hdr : packet len: 44, ttl: 254, protocol: 6 (TCP)
tcp hdr : dest port: 22, src port: 35916

------ Punt Packet Number: 3, Timestamp: 2024/07/31 01:29:48.143 ------
interface : physical: TenGigabitEthernet1/0/1[if-id: 0x00000009], pal: TenGigabitEthernet1/0/1 [if-id: 
metadata : cause: 96 [Layer2 control protocols], sub-cause: 0, q-no: 1, linktype: MCP_LINK_TYPE_LAYER2
ether hdr : dest mac: 0100.0ccc.cccc, src mac: 78bc.1a27.c203
ether hdr : length: 443
```

Si noti che il pacchetto n. 2 è il pacchetto TCP SYN da 10.10.10.1 a 10.10.20.1, in arrivo da Te1/0/9.
La nota importante è 'q-no'. Come si può notare, per passare dalla FED alla CPU viene scelta la Coda n. 14.

Qui è possibile vedere tutte le 32 code presenti per il traffico da spostare dalla FED alla CPU:

```
C9500-2#show platform hardware fed switch active qos queue stats internal cpu policer

CPU Queue Statistics
============================================================================================
(default) (set) Queue Queue
QId PlcIdx Queue Name Enabled Rate Rate Drop(Bytes) Drop(Frames)
--------------------------------------------------------------------------------------------
0 11 DOT1X Auth Yes 1000 1000 0 0
1 1 L2 Control Yes 2000 2000 0 0
2 14 Forus traffic Yes 4000 4000 0 0
3 0 ICMP GEN Yes 600 600 0 0
4 2 Routing Control Yes 5400 5400 0 0
5 14 Forus Address resolution Yes 4000 4000 0 0
6 0 ICMP Redirect Yes 600 600 0 0
7 16 Inter FED Traffic Yes 2000 2000 0 0
8 4 L2 LVX Cont Pack Yes 1000 1000 0 0
9 19 EWLC Control Yes 13000 13000 0 0
10 16 EWLC Data Yes 2000 2000 0 0
```

```
11 13 L2 LVX Data Pack Yes 1000 1000 0 0
12 0 BROADCAST Yes 600 600 0 0
13 10 Openflow Yes 200 200 0 0
14 13 Sw forwarding Yes 1000 1000 0 0
15 8 Topology Control Yes 13000 13000 0 0
16 12 Proto Snooping Yes 2000 2000 0 0
17 6 DHCP Snooping Yes 400 400 0 0
18 13 Transit Traffic Yes 1000 1000 0 0
19 10 RPF Failed Yes 200 200 0 0
20 15 MCAST END STATION Yes 2000 2000 0 0
21 13 LOGGING Yes 1000 1000 0 0
22 7 Punt Webauth Yes 1000 1000 0 0
23 18 High Rate App Yes 13000 13000 0 0
24 10 Exception Yes 200 200 0 0
25 3 System Critical Yes 1000 1000 0 0
26 10 NFL SAMPLED DATA Yes 200 200 0 0
27 2 Low Latency Yes 5400 5400 0 0
28 10 EGR Exception Yes 200 200 0 0
29 5 Stackwise Virtual OOB Yes 8000 8000 0 0
30 9 MCAST Data Yes 400 400 0 0
31 3 Gold Pkt Yes 1000 1000 0 0
```

Come si può notare, la coda n. 14 è la coda 'Inoltro software'.
In questo caso, questa coda viene utilizzata dal traffico TCP per essere puntata alla CPU.

A questo punto, è possibile acquisire una CPU (Control-Plane) su C9500-2.

Di seguito è riportata la configurazione di acquisizione della CPU:

```
C9500-2#sh mon cap test
Status Information for Capture test
Target Type:
Interface: Control Plane, Direction: BOTH
Status : Inactive
Filter Details:
Capture all packets
Buffer Details:
Buffer Type: LINEAR (default)
Buffer Size (in MB): 80
File Details:
File not associated
Limit Details:
Number of Packets to capture: 0 (no limit)
Packet Capture duration: 0 (no limit)
Packet Size to capture: 0 (no limit)
Packet sampling rate: 0 (no sampling)
C9500-2#
```

L'acquisizione viene avviata, il protocollo SSH da C9500-1 a C9200 e la cattura viene interrotta.

Di seguito sono riportati i pacchetti acquisiti dalla CPU:

```
C9500-2#show monitor capture test buffer brief
Starting the packet display ........ Press Ctrl + Shift + 6 to exit
1 0.000000 00:a3:d1:44:4b:81 -> 01:80:c2:00:00:00 STP 60 RST. Root = 32768/1/00:a3:d1:44:4b:80 Cost = 0
2 0.000010 00:a3:d1:44:4b:a3 -> 01:80:c2:00:00:00 STP 60 RST. Root = 32768/1/00:a3:d1:44:4b:80 Cost = 0
3 0.000013 00:a3:d1:44:4b:a4 -> 01:80:c2:00:00:00 STP 60 RST. Root = 32768/1/00:a3:d1:44:4b:80 Cost = 0
4 0.000016 00:a3:d1:44:4b:a6 -> 01:80:c2:00:00:00 STP 60 RST. Root = 32768/1/00:a3:d1:44:4b:80 Cost = 0
5 0.000019 00:a3:d1:44:4b:a7 -> 01:80:c2:00:00:00 STP 60 RST. Root = 32768/1/00:a3:d1:44:4b:80 Cost = 0
6 0.000022 00:a3:d1:44:4b:a8 -> 01:80:c2:00:00:00 STP 60 RST. Root = 32768/1/00:a3:d1:44:4b:80 Cost = 0
7 0.055470 c0:8b:2a:04:f0:6c -> 01:80:c2:00:00:0e LLDP 117 TTL = 120 SysName = bgl18-cx-amx-b02-2.cisco
9 0.220331 28:63:29:20:31:39 -> 00:01:22:53:74:20 0x3836 30 Ethernet II
10 0.327316 192.168.1.1 -> 224.0.0.5 OSPF 114 Hello Packet
11 0.442986 c0:8b:2a:04:f0:68 -> 01:80:c2:00:00:0e LLDP 117 TTL = 120 SysName = bgl18-cx-amx-b02-2.cisc
12 1.714121 10.10.10.1 -> 10.10.20.1 TCP 60 23098 -> 22 [SYN] Seq=0 Win=4128 Len=0 MSS=536
13 1.714375 10.10.10.1 -> 10.10.20.1 TCP 60 [TCP Out-Of-Order] 23098 -> 22 [SYN] Seq=0 Win=4128 Len=0 M
14 2.000302 00:a3:d1:44:4b:81 -> 01:80:c2:00:00:00 STP 60 RST. Root = 32768/1/00:a3:d1:44:4b:80 Cost =
15 2.000310 00:a3:d1:44:4b:a3 -> 01:80:c2:00:00:00 STP 60 RST. Root = 32768/1/00:a3:d1:44:4b:80 Cost =
~ Output Cut ~
```

Il pacchetto n. 12 è il pacchetto TCP SYN che arriva alla CPU (punt), con il valore MSS predefinito di 536.
Il pacchetto n. 13 è il pacchetto TCP SYN inviato dalla CPU (inserimento), dopo aver modificato il valore MSS in 512.

Per verificare il corretto funzionamento di questa modifica, è inoltre possibile eseguire un debug rapido della CPU.

Di seguito è riportata la configurazione di debug della CPU:

```
C9500-2#debug ip tcp adjust-mss
TCP Adjust Mss debugging is on
```

Avvio del protocollo SSH da C9500-1 a C9200:

```
C9500-1#ssh -l admin 10.10.20.1
Password:
```

Arresto del debug della CPU:

```
C9500-2#undebug all
All possible debugging has been turned off
```

Controllo dei log per i debug:

```
C9500-2#show logging
Syslog logging: enabled (0 messages dropped, 2 messages rate-limited, 0 flushes, 0 overruns, xml disabl
No Active Message Discriminator.
No Inactive Message Discriminator.
Console logging: disabled
Monitor logging: level debugging, 0 messages logged, xml disabled,
filtering disabled
Buffer logging: level debugging, 230 messages logged, xml disabled,
filtering disabled
Exception Logging: size (4096 bytes)
Count and timestamp logging messages: disabled
File logging: disabled
Persistent logging: disabled
No active filter modules.
Trap logging: level informational, 210 message lines logged
Logging Source-Interface: VRF Name:
TLS Profiles:
Log Buffer (102400 bytes):
*Jul 31 01:46:32.052: TCPADJMSS: process_enqueue_feature
*Jul 31 01:46:32.893: TCPADJMSS: process_enqueue_feature
*Jul 31 01:46:36.136: TCPADJMSS: process_enqueue_feature
*Jul 31 01:46:41.318: TCPADJMSS: process_enqueue_feature
*Jul 31 01:46:42.412: TCPADJMSS: process_enqueue_feature
*Jul 31 01:46:43.254: TCPADJMSS: process_enqueue_feature
*Jul 31 01:46:43.638: TCPADJMSS: process_enqueue_feature
*Jul 31 01:46:45.783: TCPADJMSS: Input (process)
*Jul 31 01:46:45.783: TCPADJMSS: orig_mss = 536 adj mss = 512 src_ip = 10.10.10.1 dest_ip = 10.10.20.1
*Jul 31 01:46:45.783: TCPADJMSS: paktype = 0x7F83C7BCBF78
*Jul 31 01:46:50.456: TCPADJMSS: process_enqueue_feature
*Jul 31 01:46:51.985: TCPADJMSS: process_enqueue_feature
C9500-2#
```

È possibile vedere il sovraccarico che il valore MSS originale di 536 è stato portato a 512.

Infine, è possibile acquisire un EPC su C9200 Gi1/0/3 per confermare che il pacchetto TCP SYN proviene effettivamente con un valore MSS di 512.

Di seguito è riportata la configurazione EPC:

```
C9200#sh mon cap mycap
Status Information for Capture mycap
Target Type:
Interface: GigabitEthernet1/0/3, Direction: BOTH
Status : Inactive
Filter Details:
Capture all packets
Buffer Details:
Buffer Type: LINEAR (default)
Buffer Size (in MB): 80
Limit Details:
Number of Packets to capture: 0 (no limit)
Packet Capture duration: 0 (no limit)
Packet Size to capture: 0 (no limit)
Packet sampling rate: 0 (no sampling)
C9200#
```

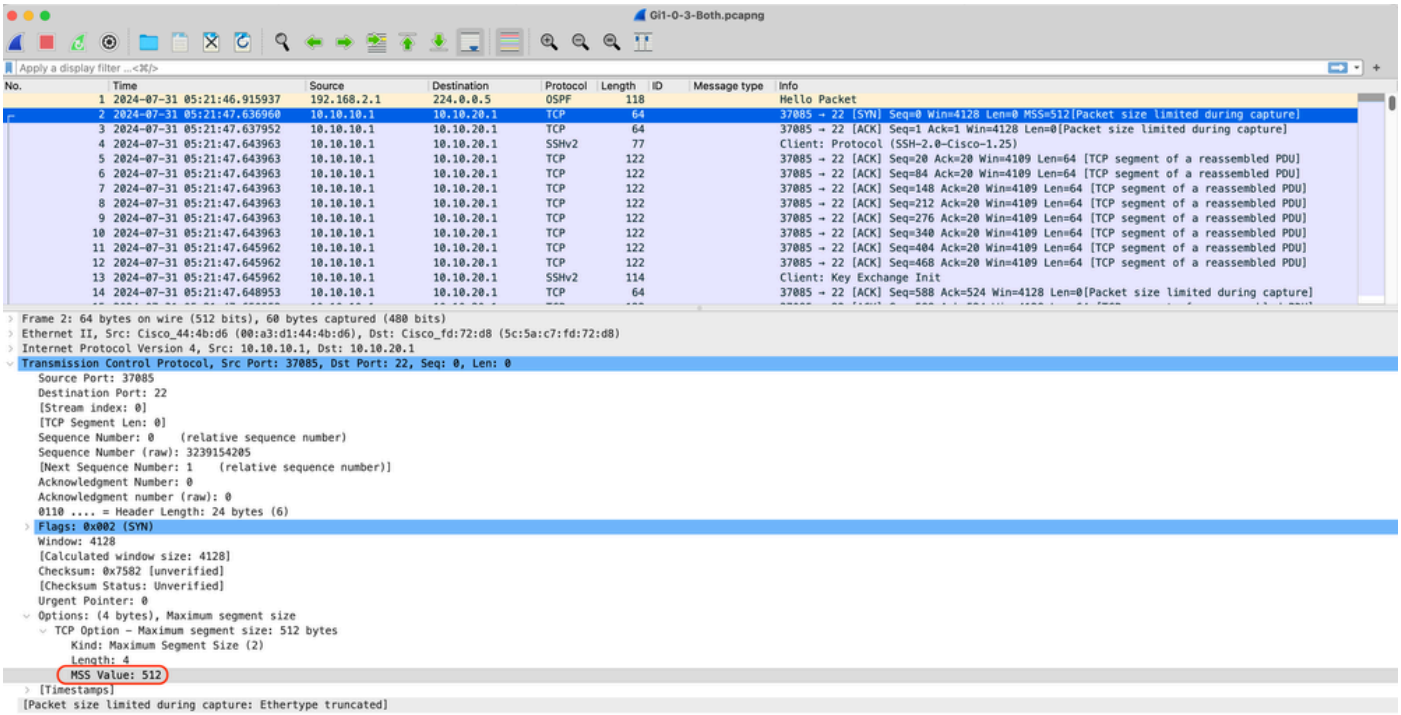
L'acquisizione viene avviata, il protocollo SSH da C9500-1 a C9200 e la cattura viene interrotta.

Di seguito sono riportati i pacchetti acquisiti dalla CPU:

```
C9200#sh mon cap mycap buff br
----------------------------------------------------------------------------
# size timestamp source destination dscp protocol
----------------------------------------------------------------------------
0 118 0.000000 192.168.2.1 -> 224.0.0.5 48 CS6 OSPF
1 64 0.721023 10.10.10.1 -> 10.10.20.1 48 CS6 TCP
2 64 0.722015 10.10.10.1 -> 10.10.20.1 48 CS6 TCP
3 77 0.728026 10.10.10.1 -> 10.10.20.1 48 CS6 TCP
4 122 0.728026 10.10.10.1 -> 10.10.20.1 48 CS6 TCP
5 122 0.728026 10.10.10.1 -> 10.10.20.1 48 CS6 TCP
6 122 0.728026 10.10.10.1 -> 10.10.20.1 48 CS6 TCP
7 122 0.728026 10.10.10.1 -> 10.10.20.1 48 CS6 TCP
8 122 0.728026 10.10.10.1 -> 10.10.20.1 48 CS6 TCP
9 122 0.728026 10.10.10.1 -> 10.10.20.1 48 CS6 TCP
10 122 0.730025 10.10.10.1 -> 10.10.20.1 48 CS6 TCP
~ Output Cut ~
```

In C9200, non è possibile visualizzare i dettagli del pacchetto come in Wireshark, ma sono disponibili solo i dettagli brevi ed esadecimali.
È quindi possibile esportare i pacchetti precedenti in un file pcap nella memoria flash.
C9200#mon cap mycap export flash:Gi1-0-3-Both.pcapng
Esportazione completata

Quindi è possibile copiare il file tramite TFTP sul PC locale e aprirlo in Wireshark.
Ecco la cattura di Wireshark.



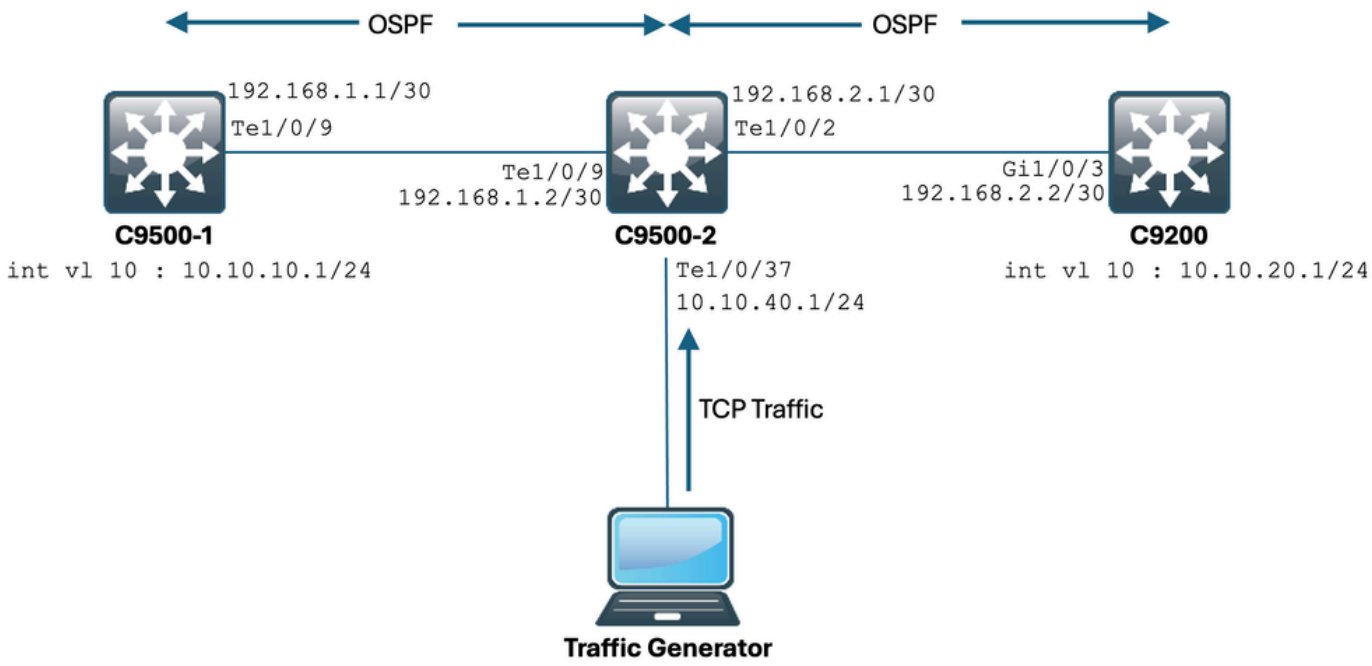
Come si può vedere, il valore TCP MSS del pacchetto SYN è 512.

Regolazione TCP MSS che provoca lentezza durante un'elevata quantità di traffico TCP

Supponiamo ora che una rete abbia più dispositivi che utilizzano il traffico TCP.
Ad esempio, possono trasferire file o accedere a un'applicazione basata su TCP (come Citrix Server).

La simulazione è stata eseguita collegando un generatore di traffico IXIA a C9500-2 Te1/0/37 e inviando pacchetti TCP SYN ad alta velocità.
Questo dispositivo IXIA funge da segmento di rete, in cui più utenti utilizzano applicazioni basate su TCP.
è stata configurata la CLI ip tcp adjust-mss su Te1/0/37.
In questo modo, tutto il traffico TCP ricevuto sul Te1/0/37 verrà indirizzato alla CPU del C9500-2.
Ciò a sua volta blocca la coda 'Inoltro software' del Policer COPP C9500-2, come accennato in precedenza nel documento.
Di conseguenza, l'istituzione della sessione SSH da C9500-1 a C9200 è compromessa.
La sessione SSH non si forma e viene timeout o viene stabilita dopo un ritardo.

Di seguito viene riportato l'aspetto della topologia:



Vediamo questo in azione.

Ecco la configurazione di C9500-2 Te1/0/37:

```
C9500-2#sh run int te1/0/37
Building configuration...
Current configuration : 135 bytes
```

```
interface TenGigabitEthernet1/0/37
no switchport
ip address 10.10.40.1 255.255.255.0
ip tcp adjust-mss 500
load-interval 30
end
```

Ora si inizia a inviare un enorme traffico da IXIA all'interfaccia Te1/0/37.
Esaminiamo ora la velocità del traffico in entrata:

```
C9500-2#sh int te1/0/37 | in rate
Queueing strategy: fifo
30 second input rate 6425812000 bits/sec, 12550415 packets/sec → We can see the enormous Input rate.
30 second output rate 0 bits/sec, 0 packets/sec
```

Provare a eseguire SSH da C9500-1 a C9200:

```
C9500-1#ssh -l admin 10.10.20.1
% Connection timed out; remote host not responding
C9500-1#
```

È evidente che il C9500-1 non è stato in grado di eseguire SSH nel C9200.
Infatti, il pacchetto TCP SYN inviato dal C9500-1 veniva scartato dalla coda 'Software Forwarding', che veniva bombardata dal traffico proveniente dal Te1/0/37.

Diamo un'occhiata alla coda:

```
C9500-2#sh platform hardware fed switch active qos queue stats internal cpu policer
CPU Queue Statistics
============================================================================================
(default) (set) Queue Queue
QId PlcIdx Queue Name Enabled Rate Rate Drop(Bytes) Drop(Frames)
--------------------------------------------------------------------------------------------
0 11 DOT1X Auth Yes 1000 1000 0 0
1 1 L2 Control Yes 2000 2000 0 0
2 14 Forus traffic Yes 4000 4000 0 0
3 0 ICMP GEN Yes 600 600 0 0
4 2 Routing Control Yes 5400 5400 0 0
5 14 Forus Address resolution Yes 4000 4000 0 0
6 0 ICMP Redirect Yes 600 600 0 0
7 16 Inter FED Traffic Yes 2000 2000 0 0
8 4 L2 LVX Cont Pack Yes 1000 1000 0 0
9 19 EWLC Control Yes 13000 13000 0 0
10 16 EWLC Data Yes 2000 2000 0 0
11 13 L2 LVX Data Pack Yes 1000 1000 0 0
12 0 BROADCAST Yes 600 600 0 0
13 10 Openflow Yes 200 200 0 0
```

```
14 13 Sw forwarding Yes 1000 1000 39683368064 620052629 → We can see the huge number of dropped packets in t
15 8 Topology Control Yes 13000 13000 0 0
16 12 Proto Snooping Yes 2000 2000 0 0
17 6 DHCP Snooping Yes 400 400 0 0
18 13 Transit Traffic Yes 1000 1000 0 0
19 10 RPF Failed Yes 200 200 0 0
20 15 MCAST END STATION Yes 2000 2000 0 0
21 13 LOGGING Yes 1000 1000 0 0
22 7 Punt Webauth Yes 1000 1000 0 0
23 18 High Rate App Yes 13000 13000 0 0
24 10 Exception Yes 200 200 0 0
25 3 System Critical Yes 1000 1000 0 0
26 10 NFL SAMPLED DATA Yes 200 200 0 0
27 2 Low Latency Yes 5400 5400 0 0
28 10 EGR Exception Yes 200 200 0 0
29 5 Stackwise Virtual OOB Yes 8000 8000 0 0
30 9 MCAST Data Yes 400 400 0 0
31 3 Gold Pkt Yes 1000 1000 0 0
```

Raccogliamo l'output più volte per garantire che il conteggio non elaborato sia in aumento durante il problema:

```
C9500-2#sh platform hardware fed switch active qos queue stats internal cpu policer | in Sw forwarding
14 13 Sw forwarding Yes 1000 1000 47046906560 735107915
14 13 21 Sw forwarding Yes
13 system-cpp-police-sw-forward : Sw forwarding/ LOGGING/ L2 LVX Data Pack/ Transit Traffic/
21 system-cpp-police-ios-feature : ICMP GEN/ BROADCAST/ ICMP Redirect/ L2 LVX Cont Pack/ Proto Snooping/
C9500-2#
!
C9500-2#sh platform hardware fed switch active qos queue stats internal cpu policer | in Sw forwarding
14 13 Sw forwarding Yes 1000 1000 47335535936 739617752
14 13 21 Sw forwarding Yes
13 system-cpp-police-sw-forward : Sw forwarding/ LOGGING/ L2 LVX Data Pack/ Transit Traffic/
21 system-cpp-police-ios-feature : ICMP GEN/ BROADCAST/ ICMP Redirect/ L2 LVX Cont Pack/ Proto Snooping/
C9500-2#
!
C9500-2#sh platform hardware fed switch active qos queue stats internal cpu policer | in Sw forwarding
14 13 Sw forwarding Yes 1000 1000 47666441088 744788145
14 13 21 Sw forwarding Yes
13 system-cpp-police-sw-forward : Sw forwarding/ LOGGING/ L2 LVX Data Pack/ Transit Traffic/
21 system-cpp-police-ios-feature : ICMP GEN/ BROADCAST/ ICMP Redirect/ L2 LVX Cont Pack/ Proto Snooping/
C9500-2#
```

Come si può vedere, il conteggio scartato è in aumento e il traffico SSH (pacchetto TCP SYN) viene scartato qui.

Se non si è consapevoli dell'interfaccia/SVI utilizzata per il traffico, è possibile usare un comando specifico.

```
C9500-2#show platform software fed switch active punt rates interfaces
Punt Rate on Interfaces Statistics
Packets per second averaged over 10 seconds, 1 min and 5 mins
```

```
=============================================================================================
| | Recv | Recv | Recv | Drop | Drop | Drop
Interface Name | IF_ID | 10s | 1min | 5min | 10s | 1min | 5min
=============================================================================================
TenGigabitEthernet1/0/37 0x00000042 1000 1000 1000 0 0 0
---------------------------------------------------------------------------------------------
C9500-2#
```

Il comando show platform software fed switch active punt rates interfaces ci fornisce l'elenco delle interfacce responsabili della ricezione di una grande quantità di traffico puntato alla CPU.
È possibile vedere chiaramente Te1/0/37 qui, che è l'interfaccia attraverso cui si ottiene il traffico TCP.

A questo punto, se si desidera visualizzare la quantità di traffico che colpisce tutte le code del Policer COPP (ricevuto sull'interfaccia precedente), è possibile utilizzare:
show platform software fed switch velocità di punt attive interfacce <IF_ID dall'output precedente>

Diamo un'occhiata:

```
C9500-2#show platform software fed switch active punt rates interfaces 0x42
Punt Rate on Single Interfaces Statistics
Interface : TenGigabitEthernet1/0/37 [if_id: 0x42]

Received Dropped
-------- -------
Total : 2048742 Total : 0
10 sec average : 1000 10 sec average : 0
1 min average : 1000 1 min average : 0
5 min average : 1000 5 min average : 0

Per CPUQ punt stats on the interface (rate averaged over 10s interval)
=============================================================================
Q | Queue | Recv | Recv | Drop | Drop |
no | Name | Total | Rate | Total | Rate |
=============================================================================
0 CPU_Q_DOT1X_AUTH 0 0 0 0
1 CPU_Q_L2_CONTROL 7392 0 0 0
2 CPU_Q_FORUS_TRAFFIC 0 0 0 0
3 CPU_Q_ICMP_GEN 0 0 0 0
4 CPU_Q_ROUTING_CONTROL 0 0 0 0
5 CPU_Q_FORUS_ADDR_RESOLUTION 0 0 0 0
6 CPU_Q_ICMP_REDIRECT 0 0 0 0
7 CPU_Q_INTER_FED_TRAFFIC 0 0 0 0
8 CPU_Q_L2LVX_CONTROL_PKT 0 0 0 0
9 CPU_Q_EWLC_CONTROL 0 0 0 0
10 CPU_Q_EWLC_DATA 0 0 0 0
11 CPU_Q_L2LVX_DATA_PKT 0 0 0 0
12 CPU_Q_BROADCAST 0 0 0 0
13 CPU_Q_CONTROLLER_PUNT 0 0 0 0
14 CPU_Q_SW_FORWARDING 2006390 1000 0 0 -----> We can see high amount of traffic hitting the Sw forward
15 CPU_Q_TOPOLOGY_CONTROL 0 0 0 0
16 CPU_Q_PROTO_SNOOPING 0 0 0 0
17 CPU_Q_DHCP_SNOOPING 0 0 0 0
18 CPU_Q_TRANSIT_TRAFFIC 0 0 0 0
19 CPU_Q_RPF_FAILED 0 0 0 0
```

```
20 CPU_Q_MCAST_END_STATION_SERVICE 0 0 0 0
21 CPU_Q_LOGGING 34960 0 0 0
22 CPU_Q_PUNT_WEBAUTH 0 0 0 0
23 CPU_Q_HIGH_RATE_APP 0 0 0 0
24 CPU_Q_EXCEPTION 0 0 0 0
25 CPU_Q_SYSTEM_CRITICAL 0 0 0 0
26 CPU_Q_NFL_SAMPLED_DATA 0 0 0 0
27 CPU_Q_LOW_LATENCY 0 0 0 0
28 CPU_Q_EGR_EXCEPTION 0 0 0 0
29 CPU_Q_FSS 0 0 0 0
30 CPU_Q_MCAST_DATA 0 0 0 0
31 CPU_Q_GOLD_PKT 0 0 0 0
------------------------------------------------------------------------
```

Raccolta dell'output più volte a intervalli molto brevi:

```
C9500-2#show platform software fed switch active punt rates interfaces 0x42 | in SW_FORWARDING
14 CPU_Q_SW_FORWARDING 2126315 1000 0 0
C9500-2#
C9500-2#show platform software fed switch active punt rates interfaces 0x42 | in SW_FORWARDING
14 CPU_Q_SW_FORWARDING 2128390 1000 0 0
C9500-2#
C9500-2#show platform software fed switch active punt rates interfaces 0x42 | in SW_FORWARDING
14 CPU_Q_SW_FORWARDING 2132295 1000 0 0
C9500-2#
```

Ciò mostra chiaramente che la coda di inoltro software è bloccata.
Dopo aver rimosso il **ip tcp adjust-mss** comando da Te1/0/37, o se si arresta il traffico TCP, l'accesso SSH da C9500-1 a C9200 viene immediatamente ristabilito.

Esaminiamo più in dettaglio la sessione SSH dopo la chiusura di C9500-2 Te1/0/37:

```
C9500-1#ssh -l admin 10.10.20.1
Password:
```

Come si può notare, l'accesso SSH è stato nuovamente ripristinato.
Pertanto, è possibile correlare la Lentezza TCP (accesso SSH bloccato) a causa dell'elevata quantità di traffico TCP nella rete, con la regolazione TCP MSS.

Punti importanti

1. Quando in rete si riscontra una lentezza del protocollo TCP, ad esempio una lentezza nel trasferimento dei file, l'accessibilità alle applicazioni correlate al protocollo TCP e così via, e la regolazione del valore TCP MSS è configurata su uno switch Catalyst, verificare le perdite del Policer COPP per controllare se la rete contiene una quantità elevata di traffico TCP.
2. Se la regolazione TCP MSS è stata configurata su uno switch Catalyst, verificare che il

traffico TCP nella rete non sovrascriva la velocità del controller COPP. In caso contrario, sulla rete potrebbero verificarsi problemi relativi al TCP (lentezza, perdita di pacchetti).

Informazioni su questa traduzione

Cisco ha tradotto questo documento utilizzando una combinazione di tecnologie automatiche e umane per offrire ai nostri utenti in tutto il mondo contenuti di supporto nella propria lingua. Si noti che anche la migliore traduzione automatica non sarà mai accurata come quella fornita da un traduttore professionista. Cisco Systems, Inc. non si assume alcuna responsabilità per l’accuratezza di queste traduzioni e consiglia di consultare sempre il documento originale in inglese (disponibile al link fornito).