

疑難排解 Catalyst 9000 系列交換器上的 MTU

目錄

[簡介](#)

[必要條件](#)

[採用元件](#)

[背景資訊](#)

[MTU摘要表](#)

[MTU問答](#)

[乙太網路框架](#)

[設定和驗證MTU](#)

[設定MTU](#)

[驗證MTU](#)

[排除MTU故障](#)

[拓撲](#)

[輸入封包捨棄 \(輸入MTU更低\)](#)

[設定和驗證IP MTU](#)

[設定IP MTU](#)

[驗證IP MTU](#)

[排除IP MTU故障](#)

[拓撲](#)

[IP分段](#)

[相關資訊](#)

[思科錯誤ID](#)

簡介

本文件說明如何瞭解和疑難排解 Catalyst 9000 系列交換器上的 MTU (最大傳輸單位) 。

必要條件


本文件沒有特定需求。

採用元件


本檔案中的資訊是根據以下硬體版本：

- C9200
- C9300
- C9400

- C9500
- C9600

 注意：您可以使用全域性命令「system mtu」同時配置裝置上所有介面的MTU大小。自 Cisco IOS® XE 17.1.1起，Catalyst 9000交換機支援每埠MTU。每埠MTU支援埠級別和埠通道級別MTU配置。使用每埠MTU，可以為不同的介面以及不同的埠通道介面設定不同的MTU值。

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。

 註：請參閱相應的配置指南，瞭解用於在其他Cisco平台上啟用這些功能的命令。

背景資訊

MTU摘要表

總幀大小= MTU + L2報頭

連線埠型別	預設MTU — 位元組	配置的MTU — 位元組	L2報頭	幀總大小
L2訪問	1500		18	1518
		9216	18	9234
L2中繼	1500		22	1522
		9216	22	9238
L3物理埠	1500		18	1518
		9216	18	9234
L3 SVI	1500		18	1518
		9216	18	9234

L3埠上的IP MTU	1500	範圍受支援	18	基於ip mtu配置值
-------------	------	-------	----	-------------

MTU問答

什麼是MTU?

- MTU是裝置可以轉發的最大傳輸單元。一般來說，此「單元」是包含IP標頭的IP封包長度。
- L2報頭（如Dot1q標籤、MacSec、SVL報頭等）未計入此計算。

L2報頭及其長度是什麼？

- 通用L2報頭是14位元組+ 4位元組的CRC，總計為18位元組
- TRUNK為dot1q vlan標籤增加4個位元組，總計22個位元組
- 同樣，MacSec將其自己的報頭長度新增到典型的L2報頭長度之上
- SVL埠新增，它自己的報頭長度位於典型的L2報頭長度之上
- 因此，線路上的整體資料包會撞到線路上

介面處理的資料包長度是多少？

- Catalyst 9000交換器處理從64位元組到9238位元組的封包大小。


什麼是預設MTU?

- 預設MTU是在任何使用者配置之前設定交換機的MTU
- 任何Catalyst 9000交換器上的預設MTU為1500位元組
- 乙太網路連線埠轉送1500位元組第3層封包+第2層標頭

MTU檢查發生在入口還是出口？

Egress:MTU是最大傳輸單位，是輸出檢查，決定分段或按原樣傳輸或捨棄的輸出

- 如果埠MTU大於要路由出去的資料包長度，則按原樣傳送資料包
- 如果封包大於輸出連線埠MTU且輸出連線埠為
 - 第3層連線埠，封包會根據MTU進行分段
 - 第2層埠，丟棄資料包。（分段僅在第3層完成）

 註：如果封包在IP標頭中設定了DF（不分段）位元，且連線埠MTU小於要路由的封包，則封包會遭捨棄

Ingress: 到達介面的封包也會執行MTU檢查

- 如果介面接收的封包超過其設定的MTU，這些封包會視為過大的封包且遭捨棄。

什麼是Jumbo資料包？

- 在Catalyst 9000交換器上，任何超過1500位元組的內容都是巨型封包或巨型封包。
 - 示例-1：如果介面MTU配置為轉發大小為9216位元組的巨型幀，則它接受或傳送大小為

9216位元組+第2層報頭的幀

- 示例-2：如果介面MTU配置為轉發大小為5000位元組的巨型幀，則它接受或傳送大小為5000位元組+第2層報頭的幀

Jumbo資料包或Oversized資料包是否被視為錯誤資料包？

- 介面通過配置的MTU丟棄收到的資料包，並將資料包報告為錯誤。
- 如果介面設定為傳輸巨量MTU，且收到的封包位於此值內，則不會將其計為錯誤。

連線埠可以處理的最小封包大小是多少？

- 64位元組（包括第2層標頭）是交換器輸入時接受的最小有效封包大小。
- 如果封包的電線長度小於64位元組，則會將其視為殘餘訊息，並在輸入訊息上捨棄。
- 如果封包應該傳輸出去，且封包少於64位元組，交換器會對封包進行填充，使其在傳輸前至少達到64位元組。

當系統MTU為9216且SVL報頭新增額外的64位元組時，會發生什麼情況？

- 第3層IP報頭下的任何報頭都不會計入MTU計算。
- SVL連結可以傳輸大小為9216 + L2標頭+ 64位元組的SVL標頭。

什麼是IP MTU？

- IP MTU僅適用於IP封包。其他非IP資料包大小不在此命令中。
- IP封包的IP MTU優先於系統MTU或每個連線埠MTU。
- IP MTU設定IP封包需要分段前可以達到的最大大小。
- 如果實體或邏輯第3層介面的MTU為1500位元組，而ip mtu為1400位元組，則無論系統或每個連線埠MTU設定如何，分段邊界均為1400位元組。
- MTU是需要與對等路由器/交換器配對的值。如果對等裝置不支援更高的MTU值，請使用IP MTU或MTU匹配兩個裝置功能。
- 當配置了IP MTU時，裝置會將路由協定資料包大小設定為配置的ip mtu值。某些路由協定依靠匹配的mtu值建立路由協定鄰居關係。
- 示例：
 - 範例1：如果介面IP MTU設定為500位元組，且介面MTU為預設值（無每個連線埠mtu），而系統MTU為9000，則介面MTU為9000位元組，IP分段為500位元組。
 - 範例2: GRE通道是輸出介面，因此24位元組的GRE標頭需要計入封包大小計算中（ip mtu 1476 + 24位元組GRE標頭= 1500總MTU）。

系統MTU和每個埠MTU有何區別？

- 系統MTU是全域性配置，用於設定整個裝置的MTU。這會將所有前面板物理埠和邏輯埠更改為system mtu命令設定的值。
- 每個埠MTU允許為每個介面設定MTU值，這優先於系統MTU配置。刪除per-port設定後，介面將回退到系統mtu。
- 示例：
 - 示例1：系統MTU值設定為9000，所有物理和邏輯埠MTU設定為9000。
 - 示例2：如果介面的MTU配置為4000，系統MTU為9000，則介面的MTU為4000，而其他埠使用MTU 9000。

由於MTU限制，分段會造成什麼影響？

- 裝置通常在資料平面中轉發已分段的資料包，但是如果裝置負責分段或重組，則可能表現出性能/資源問題。
- 分段可能對負責分段處理的應用程式和裝置的整體吞吐量和效能產生嚴重影響。
- 許多平台中的分段資料包處理在軟體中完成，需要佔用大量的cpu週期來分段或組合分段資料包。
- 如果您的網路經歷大量分段，請確保相應地調整MTU以匹配沒有分段的端對端封包流。

什麼是PMTUD (路徑MTU探索) ？

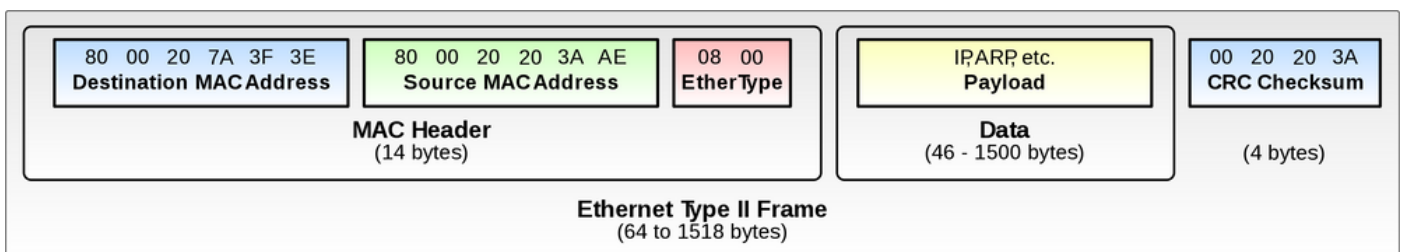
- 如上所述，TCP MSS 負責處理 TCP 連線兩個端點的分段，但不會處理這兩個端點中間存在較小 MTU 連結的情況。之所以開發出 PMTUD，是為了避免在端點之間的路徑中進行分段。它用於以動態方式判斷從封包來源到目的地沿途間的最小MTU。
- 有關PMTUD以及如何疑難排解的詳細資訊，請參閱[使用GRE和IPsec解決IPv4分段、MTU、MSS和PMTUD問題。](#)

IPv6 MTU

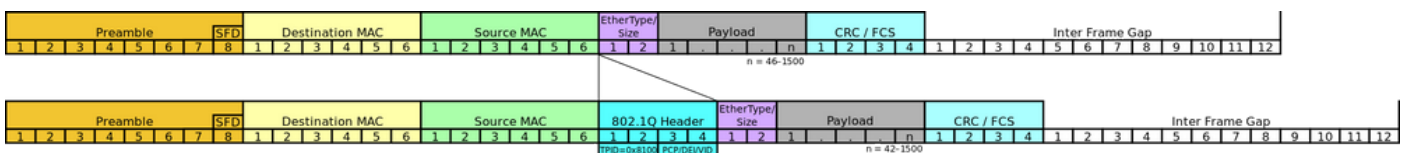
- IPv6 MTU的執行方式與IP MTU相同
- 要配置，請在介面配置下使用ipv6 mtu而不是ip mtu。
- IPv6 MTU的最小大小為1280，而IPv4為832位元組
- IPv6 PMTUD的運作方式與IPv4類似。有關詳細資訊，請參閱[IP路由配置指南、Cisco IOS® XE Amsterdam 17.3.x \(Catalyst 9500交換機 \)](#)

乙太網路框架

標準乙太網幀，無Dot1Q或其他標籤



Dot1Q乙太網幀



設定和驗證MTU

設定MTU

此配置可以全域性執行，也可以使用Cisco IOS® XE 17.1.1或更高版本在每埠級別執行。請檢查您的硬體是否支援此配置。

- 刪除埠特定的配置後，埠將使用全域性系統mtu設定

```
<#root>
```

```
### Global System MTU set to 1800 bytes ###
```

```
9500H(config)#
```

```
system mtu ?
```

```
<1500-9216> MTU size in bytes
```

```
<-- Size range that is configurable
```

```
9500H(config)#
```

```
system mtu 1800 <-- Set global to 1800 bytes
```

```
Global Ethernet MTU is set to 1800 bytes
```

```
.  
Note: this is the Ethernet payload size, not the total  
Ethernet frame size, which includes the Ethernet  
header/trailer and possibly other tags, such as ISL or  
802.1q tags.
```

```
<-- CLI provides information about what is counted as MTU
```

```
### Per-Port MTU set to 9216 bytes ###
```

```
9500H(config)#
```

```
int TwentyFiveGigE1/0/1
```

```
9500H(config-if)#
```

```
mtu 9126 <-- Interface specific MTU configuration
```

驗證MTU

本節介紹如何驗證MTU的軟體和硬體設定。

- 驗證軟體配置的MTU和硬體MTU
- 如果硬體與軟體中配置的MTU不匹配，可能會發生流量丟失

軟體MTU驗證

<#root>

```
9500H#show system mtu
Global Ethernet MTU is
1800 bytes
```

```
.
```

<-- Global level MTU

9500H#

```
show interfaces mtu
```

```
Port          Name          MTU
Twe1/0/1
```

```
9216  <-- Per-Port MTU override
```

```
Twe1/0/2
```

```
1800  <-- No per-port MTU uses global MTU
```

<...snip...>

9500H#

```
show interfaces TwentyFiveGigE 1/0/1 | inc MTU
MTU 9216
```

```
bytes, BW 1000000 Kbit/sec, DLY 10 usec,
```

9500H#

```
show interfaces TwentyFiveGigE 1/0/2 | inc MTU
MTU 1800 bytes,
```

```
BW 25000000 Kbit/sec, DLY 10 usec,
```

硬體MTU驗證

<#root>

9500H#

```
show platform software fed active ifm mappings
```

```
Interface
```

```
IF_ID
```

```
Inst Asic Core Port SubPort Mac Cntx LPN GPN Type Active
TwentyFiveGigE1/0/1
```

```
0x8
```

```
1 0 1 20 0 16 4 1 101 NIF Y
```

```
<-- Retrieve the IF_ID for use in the next command
```

```
TwentyFiveGigE1/0/2
```

```
0x9
```

```
1 0 1 21 0 17 5 2 102 NIF Y
```

```
9500H#
```

```
show platform software fed active ifm if-id 0x8 | inc MTU
```

```
Jumbo MTU .....
```


```
[9216] <-- Hardware matches software configuration
```

```
9500H#
```

```
show platform software fed active ifm if-id 0x9 | in MTU
```

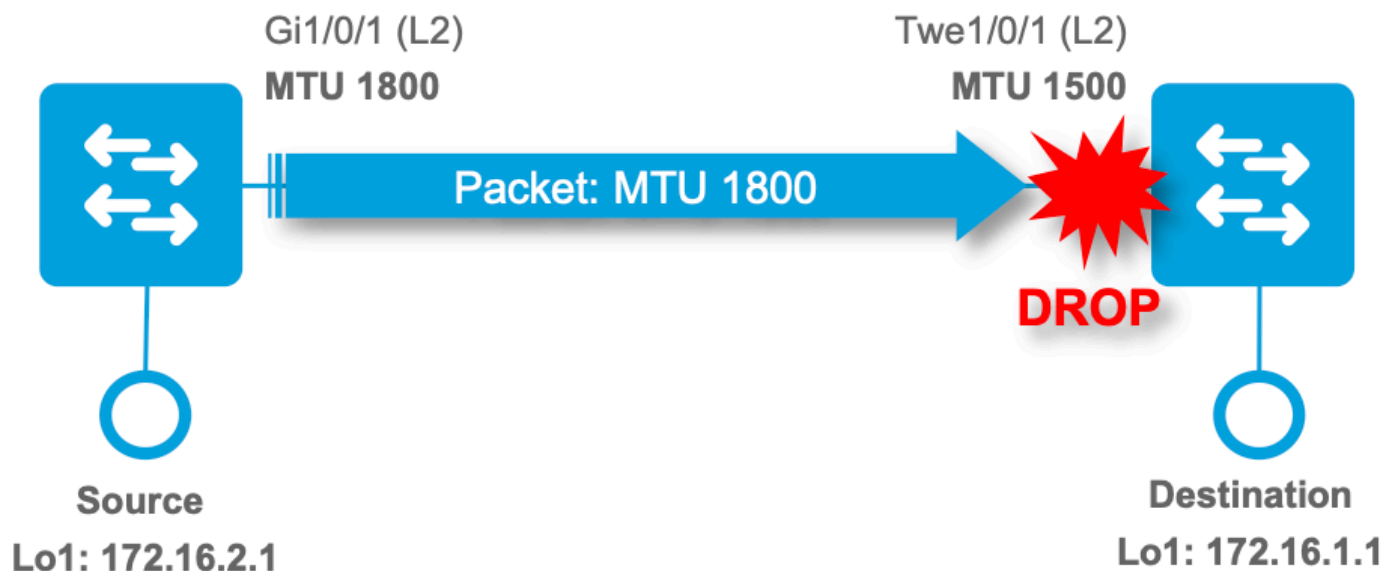
```
Jumbo MTU .....
```

```
[1800] <-- Hardware matches software configuration
```

 注意：「show platform software fed <active|standby>」可能有所不同。某些平台需要「show platform hardware fed switch <active|standby|sw_num>」

排除MTU故障

拓撲



輸入封包捨棄 (輸入MTU更低)

如果其中任一計數器遞增，則通常意味著接收的資料包已超過配置的MTU到達。

- show interface命令中的giants計數器
- 「show controller」命令中的ValidOverSize計數器

```
<#root>
```

```
9500H#
```

```
show int twentyFiveGigE 1/0/3 | i MTU  
MTU 1500 bytes,
```

```
BW 100000 Kbit/sec, DLY 100 usec,  
  0 runts,
```

```
0 giants
```

```
, 0 throttles
```

```
<-- No giants counted
```

```
9500H#
```

```
show controllers ethernet-controller twentyFiveGigE 1/0/3 | i ValidOverSize
```

```
0 Deferred frames
```

```
0 ValidOverSize frames <-- No giants counted
```

```
### 5 pings from neighbor device with MTU 1800 to ingress port MTU 1500 ###
```

```
9500H#
```

```
show int twentyFiveGigE 1/0/3 | i MTU|giant
```

```
MTU 1500 bytes, BW 100000 Kbit/sec, DLY 100 usec,  
  0 runts,
```

```
5 giants
```

```
, 0 throttles
```

```
<-- 5 giants counted
```

```
9500H#
```

```
show controllers ethernet-controller twentyFiveGigE 1/0/3 | i ValidOverSize
```

```
0 Deferred frames
```

```
5 ValidOverSize frames <-- 5 giants counted
```

關於show controllers ethernet-controller命令

- 如果資料包通過配置的MTU到達且未通過CRC檢查，則將其計為InvalidOverSize。
- 如果資料包到達配置的MTU內並且未通過CRC檢查，則將其計為FcsErr

```
<#root>
```

```
9500H#
```

```
show controllers ethernet-controller twentyFiveGigE 1/0/3 | i Fcs|InvalidOver
```

```
0 Good (>1 coll) frames
```

```
0 InvalidOverSize frames <-- MTU too large and bad CRC
```

```
0 Gold frames dropped
```

```
0 FcsErr frames          <-- MTU within limits with bad CRC
```

設定和驗證IP MTU

設定IP MTU

本節介紹如何在通道介面上設定ip mtu

- IP MTU可設定為影響本地系統產生的IP封包大小（例如路由通訊協定更新），或設定分段時的大小。

```
<#root>
```

```
C9300(config)#
```

```
interface tunnel 1
```

```
C9300(config-if)#
```

```
ip mtu 1400
```

```
interface Tunnel1
```

```
ip address 10.11.11.2 255.255.255.252
```

```
ip mtu 1400
```

```
<-- IP MTU command sets this line at 1400
```

```
ip ospf 1 area 0
```

```
tunnel source Loopback0
```

```
tunnel destination 192.168.1.1
```

驗證IP MTU

軟體IP MTU驗證

```
<#root>
```

```
C9300#
```

```
sh ip interface tunnel 1 <-- Show the IP level configuration of the interface
```

```
Tunnel1 is up, line protocol is up  
Internet address is 10.11.11.2/30  
Broadcast address is 255.255.255.255  
Address determined by setup command
```

```
MTU is 1400 bytes <-- max size of IP packet before fragmentation occurs
```

硬體IP MTU驗證

```
<#root>
```

```
C9300#sh platform software fed switch active ifm interfaces tunnel  
Interface
```

```
IF_ID
```

```
State
```

```
-----  
Tunnel1
```

```
0x00000050
```

```
READY
```

```
<-- Retrieve the IF_ID for use in the next command
```

```
C9300#sh platform software fed switch active ifm if-id 0x00000050
```

```
Interface IF_ID
```

```
: 0x0000000000000050
```

```
<-- The interface ID (IF_ID)
```

```
Interface Name : Tunnel1
```

```
Interface Block Pointer : 0x7fe98cc2d118  
Interface Block State : READY  
Interface State : Enabled  
Interface Status : ADD, UPD  
Interface Ref-Cnt : 4
```

Interface Type : TUNNEL

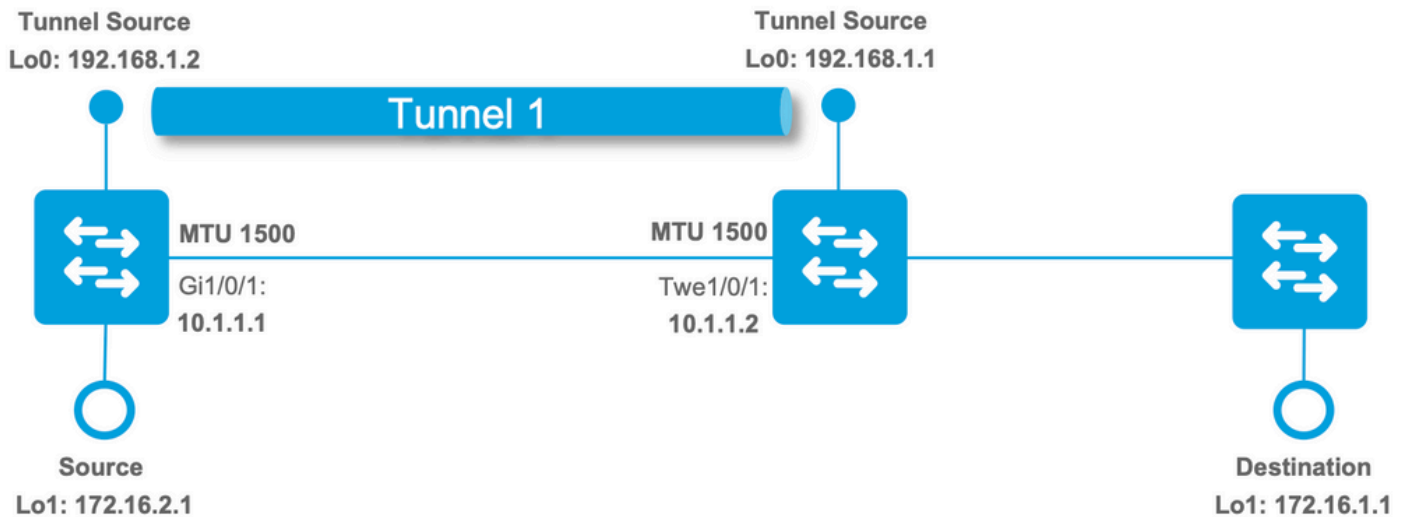
<...snip...>

Tunnel Sub-mode: 0 [none]
Hw Support : Yes
Tunnel Vrf : 0

IPv4 MTU : 1400 <-- Hardware matches software configuration
<...snip...>

排除IP MTU故障

拓撲



IP分段

透過通道介面傳送封包時，分段可以採用以下範例所述的兩種方式進行。

標準IP分段

將原始封包分段，以減少通道封裝之前的MTU。

- 只有輸入裝置負責此分段動作，片段將在實際端點而不是通道端點重新組裝
- 此類封包分段不需要消耗太多資源

<#root>

```
### Tunnel Source Device: Tunnel IP MTU 1400 | Interface MTU 1500 ###
```

C9300#

```
ping 172.16.1.1 source Loopback 1 size 1500 repeat 10 <-- ping with size over IP MTU 1400
```

```
Type escape sequence to abort.
Sending 100, 1500-byte ICMP Echos to 172.16.1.1, timeout is 2 seconds:
Packet sent with a source address of 172.16.2.1
!!!!!!!!!!!!
Success rate is 100 percent (100/100), round-trip min/avg/max = 1/1/1 ms
```

```
### Tunnel Destination Device: Ingress Capture Twel/0/1 ###
```

```
9500H#
```

```
show monitor capture 1
```

```
Status Information for Capture 1
```

```
Target Type:
```

```
Interface: TwentyFiveGigE1/0/1, Direction: IN <-- Ingress Physical interface
```

```
9500H#sh monitor capture 1 buffer br | inc IPv4|ICMP
```

```
9 22.285433 172.16.2.1 b^F^R 172.16.1.1
```

```
IPv4 1434 Fragmented IP protocol (proto=ICMP 1, off=0, ID=6c03)
```

```
10 22.285526 172.16.2.1 b^F^R 172.16.1.1 ICMP 162 Echo (ping) request id=0x0004, seq=0/0, ttl=255
```

```
11 22.286295 172.16.2.1 b^F^R 172.16.1.1
```

```
IPv4 1434 Fragmented IP protocol (proto=ICMP 1, off=0, ID=6c04)
```

```
12 22.286378 172.16.2.1 b^F^R 172.16.1.1 ICMP 162 Echo (ping) request id=0x0004, seq=1/256, ttl=2
```

```
<-- Fragmentation occurs on the Inner ICMP packet
```

```
(proto=ICMP 1)
```

```
<-- Fragments are not reassembled until they reach the actual endpoint device 172.16.1.1
```

Post通道封裝分段

將實際通道封包分段以便在封裝發生後降低MTU，但裝置檢測到MTU過大。

- 在這種情況下，通道目的地是負責片段重組的裝置，而不是真正的目的地端點
- 發生組態問題時，會發生這種情況。裝置設定的IP MTU高於應用隧道報頭後實際埠或系統MTU可以處理的大小。
- 在這種情況下，通道來源必須分割通道本身，而通道目的地必須重組通道標頭，才能將封包傳送到下一個躍點或目的地。
- 這種報頭分段可能增加大量的處理開銷；這取決於必須處理的流的速率。

- 根據平台、代碼和流量速率，您還可以看到CoPP類「Forus traffic」中的資料包丟失和丟棄

```
<#root>
```

```
### Tunnel Source Device: Tunnel IP MTU 1500 | Interface MTU 1500 ###
```

```
C9300(config-if)#
```

```
ip mtu 1500
```

```
%Warning: IP MTU value set 1500 is greater than the current transport value 1476, fragmentation may occur  
<-- Device warns the user that this can cause fragmentation (this is a configuration issue)
```

```
### Tunnel Destination Device: Ingress Capture Twe1/0/1 ###
```

```
9500H#
```

```
show monitor capture 1
```

```
Status Information for Capture 1  
Target Type:
```

```
Interface: TwentyFiveGigE1/0/1, Direction: IN <-- Ingress Physical interface
```

```
9500H
```

```
#sh monitor capture 1 buffer br | i IPv4|ICMP
```

```
1 0.000000
```

```
192.168.1.2 b^F^R 192.168.1.1
```

```
IPv4 1514 Fragmented IP protocol (proto=Generic Routing Encapsulation 47
```

```
, off=0, ID=4501)
```

```
2 0.000042 172.16.2.1 b^F^R 172.16.1.1 ICMP 60 Echo (ping) request id=0x0005, seq=0/0, ttl=255
```

```
3 2.000598
```

```
192.168.1.2 b^F^R 192.168.1.1
```

```
IPv4 1514 Fragmented IP protocol (proto=Generic Routing Encapsulation 47
```

```
, off=0, ID=4502)
```

```
4 2.000642 172.16.2.1 b^F^R 172.16.1.1 ICMP 60 Echo (ping) request id=0x0005, seq=1/256, ttl=255
```

```
<-- Fragmentation has occurred on the outer GRE header(proto=Generic Routing Encapsulation 47)
```

```
<-- Fragments must be reassembled at the Tunnel endpoint, in this case the 9500
```

相關資訊

- [技術支援與文件 - Cisco Systems](#)
- [介面和硬體元件配置指南,Cisco IOS® XE Amsterdam 17.3.x \(Catalyst 9500交換機 \)](#)
- [介面和硬體元件配置指南,Cisco IOS® XE Amsterdam 17.3.x \(Catalyst 9600交換機 \)](#)
- [使用 GRE 和 IPsec 解決 IPv4 分段、MTU、MSS 和 PMTUD 問題](#)

思科錯誤ID

重新載入後不受Cisco錯誤ID [CSCvr84911](#)系統MTU

思科錯誤ID [CSCvq30464](#)CAT9400:MTU配置未應用於處於活動狀態的非活動埠

思科錯誤 ID [重新載入](#)後不會考慮CSCvh04282 Cat9300非預設系統MTU設定值

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）。