

# Cisco IOS XR和Cisco IOS路由器上的MTU行為

## 目錄

[簡介](#)

[背景資訊](#)

[設定](#)

[Cisco IOS與Cisco IOS XR軟體的比較](#)

[路由的L3介面](#)

[預設MTU](#)

[非預設MTU](#)

[路由的L3子介面](#)

[L2VPN L2介面](#)

[EVC\(ASR9000\)](#)

[非EVC\(XR 12000和CRS\)](#)

[自動乙太網路介面驅動程式MTU和MRU配置](#)

[從低於5.1.1的版本升級到5.1.1或更高版本時，轉換配置](#)

## 簡介

本檔案介紹Cisco IOS® XR路由器上的最大傳輸單元(MTU)行為，並將這些行為與Cisco IOS路由器進行比較。它還討論了路由第3層(L3)介面和第2層VPN(L2VPN)第2層介面上同時使用乙太網路虛擬連線(EVC)和非EVC模型的MTU。本檔案還說明在5.1.1版及更新版本中自動設定乙太網路介面驅動程式MTU和最大接收單元(MRU)的方式的重要變更。

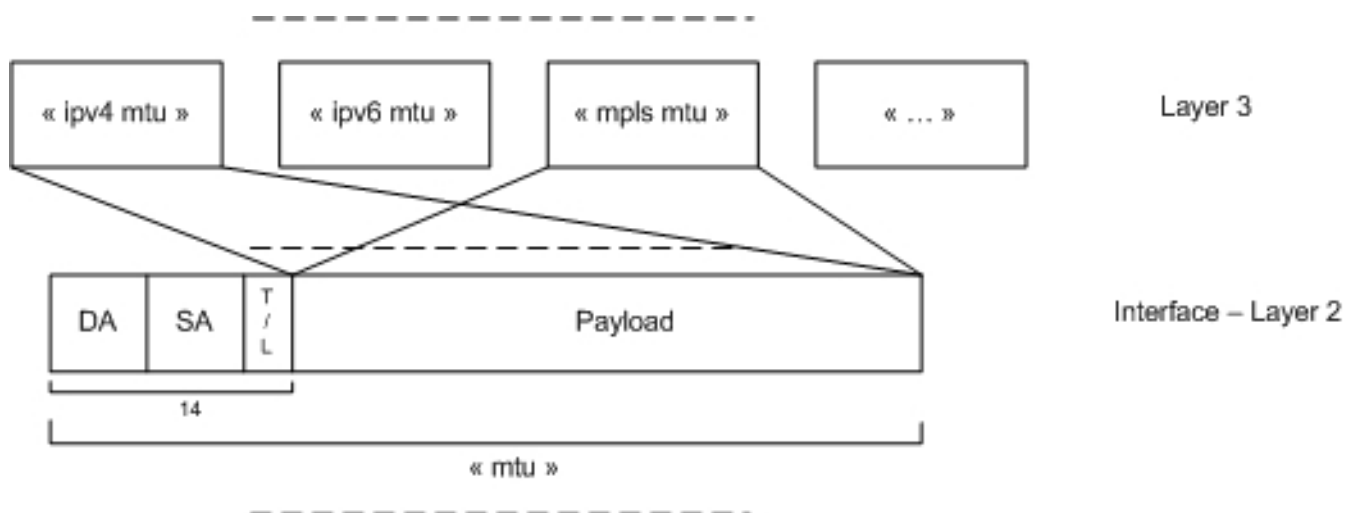
## 背景資訊

在電腦網路中，一個層的通訊協定的MTU定義了允許該層通過一個介面傳輸的最大協定資料單元的大小（以位元組為單位）。一個MTU引數與每個介面、層和協定相關聯。

Cisco IOS XR軟體中的MTU特點如下：

- 位於L2和L3的MTU **config**和**show**命令包括其相應層的報頭大小。例如，配置L2 MTU的**mtu**命令包括14個位元組的乙太網路介面（不帶dot1q），或者4個位元組的點對點協定(PPP)或高級資料鏈路控制(HDLC)。 **ipv4 mtu**命令包括20位元組的IPv4標頭。
- 較高層的MTU必須適合較低層的負載中。例如，如果非dot1q乙太網路介面的介面MTU是預設的1514位元組，則較高層通訊協定(例如多重協定標籤交換(MPLS))在該介面上最多可以有1500位元組。這意味著您只能將1500位元組的MPLS幀（包括標籤）放入乙太網路幀中。如果您希望在1500位元組IPv4封包之上允許兩個MPLS標籤，則無法在該介面上設定1508位元組的MPLS MTU。為了在乙太網路介面上傳輸1508位元組的MPLS幀，介面MTU必須增加到1522或更

高的值，以確保第2層介面負載足夠大，可以承載MPLS幀。



- 在經典的Cisco IOS軟體（而不是Cisco IOS XR軟體）中，`interface mtu`指令會設定L2負載大小，但並不包括L2標頭。這與Cisco IOS XR軟體不同，後者在`interface mtu`指令中包含L2和L3額外負荷。第3層MTU命令(如`ipv4 mtu`命令的情況)會設定該通訊協定（包括L3標頭）的最大封包大小。這類似於Cisco IOS XR軟體。
- Cisco IOS XR軟體中的預設介面MTU必須允許傳輸1500位元組的L3封包。因此，主以太網介面的預設MTU為1514位元組，串列介面的預設MTU為1504位元組。

本文檔的其餘部分說明MTU特徵，比較Cisco IOS和Cisco IOS XR軟體行為，並提供以下型別介面的示例：

- 路由的L3介面
- 路由的L3子介面
- L2VPN L2介面

## 設定

註：使用[命令查詢工具](#)(僅限[註冊](#)客戶)可獲取本節中使用的命令的詳細資訊。

註：[Output Interpreter Tool](#)(僅供已註冊客戶)支援某些`show`命令。使用Output Interpreter工具檢視`show`指令輸出的分析。

## Cisco IOS與Cisco IOS XR軟體的比較

本節將比較Cisco IOS和Cisco IOS XR軟體行為與參考MTU特徵。

在Cisco IOS軟體中，`mtu`指令和對應的`show`指令不包括L2標頭。使用`mtu`命令可將L2負載設定為L3封包（包括L3標頭）的最大大小。

這與Cisco IOS XR軟體不同，Cisco IOS XR軟體中`mtu`命令包括L2標頭（以太網路為14位元組，PPP/HDLC為4位元組）。

如果Cisco IOS路由器配置了mtu x並連線到Cisco IOS XR路由器，則應該為乙太網介面配置mtu x+14，為串列介面配置mtu x+4。

Cisco IOS和Cisco IOS XR軟體對ipv4 mtu、ipv6 mtu和mpls mtu命令具有相同的含義；必須使用相同的值配置它們。

因此，這是乙太網介面上Cisco IOS軟體中的配置：

```
mtu 9012
ipv4 mtu 9000
ipv6 mtu 9000
```

Cisco IOS XR軟體鄰居上的對應組態為：

```
mtu 9026
ipv4 mtu 9000
ipv6 mtu 9000
```

## 路由的L3介面

連線到第2層網路的所有裝置的MTU值必須相同。否則，可能會報告以下症狀：

- 中間系統到中間系統(IS-IS)鄰接關係不會出現。預設情況下，IS-IS使用hello-padding；因此，hello可能會被描述為巨型，當一台路由器的MTU值低於其他路由器的值時，可能會被丟棄。
- 開放最短路徑優先(OSPF)鄰接關係會停滯在Exstart或Exchange狀態，因為大型資料庫描述符(DBD)資料包可能被視為巨型資料包，可能會被丟棄。當路由器上接收到具有較低MTU值的資料包時，資料庫不會同步。
- 資料流量的特徵為巨型資料流量，當資料流量在傳輸裝置上接收時，資料流量的MTU值小於傳輸裝置上的MTU值，資料流量將被丟棄。
- 大型封包遭捨棄時，輸送量較低。在路徑MTU探索的情況下，TCP作業階段會在捨棄大型封包時復原，但會影響輸送量。

## 預設MTU

本節分析未設定mtu指令時的路由介面的預設MTU:

```
RP/0/RP0/CPU0:motorhead#sh run int gigabitEthernet 0/1/0/3
interface GigabitEthernet0/1/0/3
 cdp
 ipv4 address 10.0.1.1 255.255.255.0
 ipv6 address 2001:db8::1/64
!

RP/0/RP0/CPU0:router#sh int gigabitEthernet 0/1/0/3 | i MTU
MTU 1514 bytes, BW 1000000 Kbit (Max: 1000000 Kbit)
RP/0/RP0/CPU0:router#show im database interface gigabitEthernet 0/1/0/3

View: OWN - Owner, L3P - Local 3rd Party, G3P - Global 3rd Party,
LDP - Local Data Plane, GDP - Global Data Plane, RED - Redundancy

Node 0/1/CPU0 (0x11)

Interface GigabitEthernet0/1/0/3, ifh 0x01180100 (up, 1514)
```

```

Interface flags: 0x00000000010059f (IFCONNECTOR|IFINDEX
|SUP_NAMED_SUB|BROADCAST|CONFIG|HW|VIS|DATA
|CONTROL)
Encapsulation: ether
Interface type: IFT_ETHERNET
Control parent: None
Data parent: None
Views: GDP|LDP|L3P|OWN

```

```

Protocol Caps (state, mtu)
-----
None ether (up, 1514)
arp arp (up, 1500)
clns clns (up, 1500)
ipv4 ipv4 (up, 1500)
mpls mpls (up, 1500)
ipv6 ipv6_preswitch (up, 1500)
ipv6 ipv6 (down, 1500)
ether_sock ether_sock (up, 1500)

```

```

RP/0/RP0/CPU0:router#show ipv4 interface gigabitEthernet 0/1/0/3 | i MTU
MTU is 1514 (1500 is available to IP)

```

```

RP/0/RP0/CPU0:router#show ipv6 interface gigabitEthernet 0/1/0/3 | i MTU
MTU is 1514 (1500 is available to IPv6)

```

```

RP/0/RP0/CPU0:router#sh mpls interfaces gigabitEthernet 0/1/0/3 private location 0/1/CPU0
Interface IFH MTU

```

```

-----
Gi0/1/0/3 0x01180100 1500

```

```

RP/0/RP0/CPU0:router#

```

在本示例中，預設的L2介面MTU為1514位元組，包括14位元組的乙太網報頭。14位元組包含目的地MAC位址的6位元組、來源MAC位址的6位元組以及型別或長度的2位元組。這不包括前導碼、幀分隔符、4位元組的幀校驗序列(FCS)和幀間間隙。對於PPP或HDLC幀，將計入第2層報頭的4個位元組；因此預設介面MTU為1504位元組。

L3子協定從父MTU的負載繼承其MTU。從1514位元組的L2 MTU減去L2報頭的14位元組時，L2負載為1500位元組。這會成為第3層通訊協定的MTU。IPv4、IPv6、MPLS和無連線網路服務(CLNS)繼承此1500位元組MTU。因此，Cisco IOS XR乙太網路介面預設可以傳輸與Cisco IOS乙太網路介面上的預設值相同的1500位元組第3層封包。

## 非預設MTU

本節介紹如何設定mpls mtu 1508，以便傳送1500位元組的IPv4封包，其中在封包之上有兩個每個4位元組的MPLS標籤：

```

RP/0/RP0/CPU0:router#conf
RP/0/RP0/CPU0:router(config)#int gig 0/1/0/3
RP/0/RP0/CPU0:router(config-if)#mpls mtu 1508
RP/0/RP0/CPU0:router(config-if)#commit
RP/0/RP0/CPU0:Mar 12 00:36:49.807 CET: config[65856]: %MGBL-CONFIG-6-DB_COMMIT : Configuration
committed by user 'root'. Use 'show configuration commit changes 1000000124' to view the
changes.RP/0/RP0/CPU0:router(config-if)#end
RP/0/RP0/CPU0:Mar 12 00:36:54.188 CET: config[65856]: %MGBL-SYS-5-CONFIG_I : Configured
from console by root on vty0 (10.55.144.149)
RP/0/RP0/CPU0:router#sh mpls interfaces gigabitEthernet 0/1/0/3 private location 0/1/CPU0
Interface IFH MTU
-----
Gi0/1/0/3 0x01180100 1500
RP/0/RP0/CPU0:router#show im database interface gigabitEthernet 0/1/0/3

```

View: OWN - Owner, L3P - Local 3rd Party, G3P - Global 3rd Party,  
LDP - Local Data Plane, GDP - Global Data Plane, RED - Redundancy

Node 0/1/CPU0 (0x11)

Interface GigabitEthernet0/1/0/3, ifh 0x01180100 (up, 1514)

Interface flags: 0x00000000010059f (IFCONNECTOR|IFINDEX  
|SUP\_NAMED\_SUB|BROADCAST|CONFIG|HW|VIS|DATA  
|CONTROL)

Encapsulation: ether

Interface type: IFT\_ETHERNET

Control parent: None

Data parent: None

Views: GDP|LDP|L3P|OWN

Protocol Caps (state, mtu)

-----

None ether (up, 1514)

arp arp (up, 1500)

clns clns (up, 1500)

ipv4 ipv4 (up, 1500)

mpls mpls (up, **1500**)

ipv6 ipv6\_preswitch (up, 1500)

ipv6 ipv6 (down, 1500)

ether\_sock ether\_sock (up, 1500)

RP/0/RP0/CPU0:router#

雖然已提交 **mpls mtu 1508** 命令，但並未應用該命令，因為 MPLS 在 **show** 命令中仍具有 1500 位元組的 MTU。這是因為 L3 子協定的 MTU 不能大於其父 L2 介面的負載。

為了在 1500 位元組的 IP 封包上允許兩個標籤，您必須：

- 將 L2 介面 MTU 配置為 1522 位元組，以便所有子協定（包括 MPLS）繼承 1508 位元組的 MTU (1522 - 14 = 1508)。
- 將第 3 層協定的 MTU 降低到 1500 位元組，以便僅允許 MPLS 超過 1500 位元組。

RP/0/RP0/CPU0:router#**sh run int gig 0/1/0/3**

interface GigabitEthernet0/1/0/3

cdp

**mtu 1522**

**ipv4 mtu 1500**

ipv4 address 10.0.1.1 255.255.255.0

**ipv6 mtu 1500**

ipv6 address 2001:db8::1/64

!

!

RP/0/RP0/CPU0:router#**show im database interface gigabitEthernet 0/1/0/3**

View: OWN - Owner, L3P - Local 3rd Party, G3P - Global 3rd Party,  
LDP - Local Data Plane, GDP - Global Data Plane, RED - Redundancy

Node 0/1/CPU0 (0x11)

Interface GigabitEthernet0/1/0/3, ifh 0x01180100 (up, **1522**)

Interface flags: 0x00000000010059f (IFCONNECTOR|IFINDEX  
|SUP\_NAMED\_SUB|BROADCAST|CONFIG|HW|VIS|DATA  
|CONTROL)

Encapsulation: ether

```
Interface type: IFT_ETHERNET
Control parent: None
Data parent: None
Views: GDP|LDP|L3P|OWN
```

```
Protocol Caps (state, mtu)
-----
None ether (up, 1522)
arp arp (up, 1508)
clns clns (up, 1508)
ipv4 ipv4 (up, 1500)
mpls mpls (up, 1508)
ipv6 ipv6_preswitch (up, 1508)
ipv6 ipv6 (down, 1500)
ether_sock ether_sock (up, 1508)
```

```
RP/0/RP0/CPU0:router#
```

此組態允許您傳送1500位元組的IPv4和IPv6封包和1508位元組的MPLS封包（頂部有兩個標籤的1500位元組封包）。

## 路由的L3子介面

這些特性適用於路由的第3層子介面。

路由的子介面MTU繼承其父主介面的MTU；為子介面上配置的每個VLAN標籤新增4個位元組。因此，dot1q子介面有4個位元組，IEEE 802.1Q隧道(QinQ)子介面有8個位元組。

因此，可以在主介面和子介面上轉發相同大小的L3資料包。

可以在子介面下配置mtu命令，但僅當它小於或等於從主介面繼承的MTU時才應用。

以下範例中，主介面的MTU為2000位元組：

```
RP/0/RP0/CPU0:router#sh run int gig 0/1/0/3
interface GigabitEthernet0/1/0/3
cdp
mtu 2000
!
```

```
RP/0/RP0/CPU0:router#sh run int gig 0/1/0/3.100
interface GigabitEthernet0/1/0/3.100
ipv4 address 10.0.2.1 255.255.255.0
ipv6 address 2001:db9:0:1::1/64
dot1q vlan 100
!
```

```
RP/0/RP0/CPU0:router#sh int gig 0/1/0/3.100 | i MTU
MTU 2004 bytes, BW 1000000 Kbit (Max: 1000000 Kbit)
RP/0/RP0/CPU0:router#show im database interface gigabitEthernet 0/1/0/3.100
```

```
View: OWN - Owner, L3P - Local 3rd Party, G3P - Global 3rd Party,
LDP - Local Data Plane, GDP - Global Data Plane, RED - Redundancy
```

```
Node 0/1/CPU0 (0x11)
```

```
Interface GigabitEthernet0/1/0/3.100, ifh 0x01180260 (up, 2004)
Interface flags: 0x0000000000000597 (IFINDEX|SUP_NAMED_SUB
```

```
|BROADCAST|CONFIG|VIS|DATA|CONTROL)
Encapsulation: dot1q
Interface type: IFT_VLAN_SUBIF
Control parent: GigabitEthernet0/1/0/3
Data parent: GigabitEthernet0/1/0/3
Views: GDP|LDP|L3P|OWN
```

```
Protocol Caps (state, mtu)
-----
None vlan_jump (up, 2004)
None dot1q (up, 2004)
arp arp (up, 1986)
ipv4 ipv4 (up, 1986)
ipv6 ipv6_preswitch (up, 1986)
ipv6 ipv6 (down, 1986)
```

```
RP/0/RP0/CPU0:router#
```

在show命令中，子介面的MTU為2004；由於子介面下設定了一個dot1q標籤，因此主介面的MTU需增加4位元組。

但是，IPv4和IPv6資料包的MTU仍然與主介面(1986)的MTU相同。這是因為第3層通訊協定的MTU現在計算為： $2004 - 14 - 4 = 1986$ 。

可以在子介面下配置mtu命令，但配置的MTU僅在它小於或等於從主介面繼承的MTU（比主介面的MTU大4位元組）時才應用。

當子介面的MTU大於繼承的MTU時，不會應用它，如下所示：

```
RP/0/RP0/CPU0:router#sh int gig 0/1/0/3.100 | i MTU
MTU 2004 bytes, BW 1000000 Kbit (Max: 1000000 Kbit)
RP/0/RP0/CPU0:router#conf
RP/0/RP0/CPU0:router(config)#int gig 0/1/0/3.100
RP/0/RP0/CPU0:router(config-subif)#mtu 2100
RP/0/RP0/CPU0:router(config-subif)#commit
RP/0/RP0/CPU0:router(config-subif)#end
RP/0/RP0/CPU0:router#sh int gig 0/1/0/3.100 | i MTU
MTU 2004 bytes, BW 1000000 Kbit (Max: 1000000 Kbit)
RP/0/RP0/CPU0:router#
```

因此，您只能使用mtu命令來降低從主介面繼承的MTU值。

同樣地，您還可以使用第3層通訊協定(IPv4、IPv6、MPLS)的MTU指令，降低從子介面L2負載繼承的第3層MTU值。當將第3層通訊協定MTU設定為不適合L2 MTU負載的值時，MTU不會生效。

## L2VPN L2介面

L2VPN的MTU非常重要，因為當PW兩端的連線電路上的MTU不同時，標籤發佈通訊協定(LDP)不會啟動偽線(PW)。

以下是show命令，說明L2VPN PW在MTU不相符時停機：

```
RP/0/RP0/CPU0:router1#sh l2vpn xconnect
Legend: ST = State, UP = Up, DN = Down, AD = Admin Down, UR = Unresolved,
SB = Standby, SR = Standby Ready, (PP) = Partially Programmed

XConnect Segment 1 Segment 2
```

```

Group Name ST Description ST Description ST
-----
mtu mtu DN Gi0/0/0/2.201 UP 10.0.0.12 201 DN
-----
RP/0/RP0/CPU0:router1#sh l2vpn xconnect detail

```

```

Group mtu, XC mtu, state is down; Interworking none
AC: GigabitEthernet0/0/0/2.201, state is up
Type VLAN; Num Ranges: 1
VLAN ranges: [201, 201]
MTU 2000; XC ID 0x1080001; interworking none
Statistics:
packets: received 0, sent 0
bytes: received 0, sent 0
drops: illegal VLAN 0, illegal length 0
PW: neighbor 10.0.0.12, PW ID 201, state is down ( local ready )
PW class mtu-class, XC ID 0xffffe0001
Encapsulation MPLS, protocol LDP
Source address 10.0.0.2
PW type Ethernet, control word disabled, interworking none
PW backup disable delay 0 sec
Sequencing not set

```

```

PW Status TLV in use
MPLS Local Remote
-----

```

```

Label 16046 16046
Group ID 0x1080100 0x6000180
Interface GigabitEthernet0/0/0/2.201 GigabitEthernet0/1/0/3.201
MTU 2000 1986
Control word disabled disabled
PW type Ethernet Ethernet
VCCV CV type 0x2 0x2
(LSP ping verification) (LSP ping verification)
VCCV CC type 0x6 0x6
(router alert label) (router alert label)
(TTL expiry) (TTL expiry)
-----

```

```

Incoming Status (PW Status TLV):
Status code: 0x0 (Up) in Notification message
Outgoing Status (PW Status TLV):
Status code: 0x0 (Up) in Notification message
MIB cpwVcIndex: 4294836225
Create time: 18/04/2013 16:20:35 (00:00:37 ago)
Last time status changed: 18/04/2013 16:20:43 (00:00:29 ago)
Error: MTU mismatched
Statistics:
packets: received 0, sent 0
bytes: received 0, sent 0

```

```

RP/0/RP0/CPU0:router1#
RP/0/RP0/CPU0:router1#sh int GigabitEthernet0/0/0/2 | i MTU
MTU 2014 bytes, BW 1000000 Kbit (Max: 1000000 Kbit)
RP/0/RP0/CPU0:router1#sh int GigabitEthernet0/0/0/2.201 | i MTU
MTU 2018 bytes, BW 1000000 Kbit (Max: 1000000 Kbit)
RP/0/RP0/CPU0:router1#

```

在此示例中，請注意，每端的MPLS L2VPN提供商邊緣(PE)必須發出相同的MTU值才能啟用PW。

由MPLS LDP發出訊號的MTU不包括L2開銷。這不同於包括L2開銷的XR interface config和show命令。子介面上的MTU為2018位元組（繼承自2014位元組的主介面），但LDP發出的MTU為2000位元組。因此，它從L2報頭中減去18位元組（14位元組的乙太網報頭+ 4位元組的1 dot1q標籤）。

瞭解每台裝置如何計算連線電路的MTU值以修復MTU不匹配非常重要。這取決於供應商、平台、軟



體版本和配置等引數。

## EVC(ASR9000)

Cisco ASR 9000系列聚合服務路由器使用EVC基礎設施模型，允許在L2VPN L2介面和子介面上靈活地匹配VLAN。

EVC L2VPN L2介面具有以下特性：

- 它們允許使用**encapsulation**命令設定一個或多個標籤。
- 預設情況下，僅使用**encapsulation**命令時，標籤會被保留並通過PW傳輸。因此，在預設情況下，您不需要像在非EVC平台上那樣去除標籤。
- 當您決定彈出傳入標籤或把一些附加標籤推到傳入幀頂部時，請使用**rewrite**命令。

若要計運算元介面MTU，請採用主介面MTU（預設值或主介面上手動設定的值），並為使用**encapsulation**指令設定的每個VLAN標籤新增4位元組。請參閱[特定的EFP封裝命令](#)。

當子介面上有**mtu**命令時，它僅在低於計算MTU時生效。**rewrite**指令不會影響子介面的MTU。

以下是範例：

```
RP/0/RSP0/CPU0:router2#sh run int gig 0/1/0/3
interface GigabitEthernet0/1/0/3
cdp
mtu 2014
negotiation auto
!
```

```
RP/0/RSP0/CPU0:router2#sh run int gig 0/1/0/3.201
interface GigabitEthernet0/1/0/3.201 l2transport
encapsulation dot1q 201 second-dot1q 10
rewrite ingress tag pop 2 symmetric
!
```

```
RP/0/RSP0/CPU0:router2#
RP/0/RSP0/CPU0:router2#sh int gig 0/1/0/3.201
GigabitEthernet0/1/0/3.201 is up, line protocol is up
Interface state transitions: 1
Hardware is VLAN sub-interface(s), address is 0024.986c.63f3
Layer 2 Transport Mode
MTU 2022 bytes, BW 1000000 Kbit (Max: 1000000 Kbit)
```

在本例中，主介面的MTU為2014位元組；新增8位元組，因為子介面上配置了兩個標籤。

如果在子介面上配置**mtu 2026 bytes**，則由於它大於從主介面(2022)繼承的子介面的MTU，所以不會應用它。因此，只能配置小於2022位元組的子介面MTU。

基於此子介面MTU，計算向鄰居發出訊號的MPLS LDP負載的MTU，並確保它與遠端L2VPN PE計算的負載相同。**rewrite**指令在此發揮作用。

為了計算MPLS LDP淨荷的MTU，取子介面的MTU，然後：

1. 為乙太網報頭減去14個位元組。
2. 為子介面上配置的**rewrite**命令中彈出的每個標籤減去4個位元組。
3. 為子介面上配置的**rewrite**命令中推送的每個標籤新增4個位元組。

在gig 0/1/0/3.201上使用QinQ配置時，會出現以下相同示例：

```
interface GigabitEthernet0/1/0/3
cdp
mtu 2014
negotiation auto
!
interface GigabitEthernet0/1/0/3.201 l2transport
encapsulation dot1q 201 second-dot1q 10
rewrite ingress tag pop 2 symmetric
!

RP/0/RSP0/CPU0:router2#sh int gig 0/1/0/3.201
GigabitEthernet0/1/0/3.201 is up, line protocol is up
Interface state transitions: 1
Hardware is VLAN sub-interface(s), address is 0024.986c.63f3
Layer 2 Transport Mode
MTU 2022 bytes, BW 1000000 Kbit (Max: 1000000 Kbit)
```

以下是MPLS LDP負載MTU的計算：

1. 子介面MTU的MTU值：2022位元組
2. 減去14位元組的乙太網路標頭：2022 - 14 = 2008位元組
3. 減去4個位元組，表示在 **重寫** 命令：2008 - 4 \* 2 = 2000

確保遠端通告2000位元組的MPLS LDP負載。否則，請調整本地或遠端連線電路(AC)的MTU大小，使其匹配。

```
RP/0/RSP0/CPU0:router2#sh l2vpn xconnect det

Group mtu, XC mtu, state is up; Interworking none
AC: GigabitEthernet0/1/0/3.201, state is up
Type VLAN; Num Ranges: 1
Outer Tag: 201
VLAN ranges: [10, 10]
MTU 2000; XC ID 0x1880003; interworking none
```

### 特定乙太網路流量點(EFP)封裝命令

這些封裝會計為匹配的零標籤，因此不會增加子介面MTU：

- 封裝未標籤
- 預設封裝

這些封裝修改器不會影響計運算元介面MTU所需的標籤數：

- 本機
- payload-ethertype
- 精確
- COS
- ingress source-mac或ingress destination-mac

**encapsulation [dot1q|dot1ad]優先順序標籤的計數與單個標籤匹配。**

用作最內部標籤匹配的「any」關鍵字不會增加子介面MTU。

- **encapsulation dot1q any**不會增加子介面MTU。

- `encapsulation dot1ad 10 dot1q any`作為一個標籤；它將子介面MTU增加4個位元組。
- `encapsulation dot1ad any dot1q 7`被視為兩個標籤；它將子介面MTU增加8個位元組。

VLAN-ID的範圍會增加子介面的MTU:

- `encapsulation dot1q 10-100`作為一個標籤；它將子介面MTU增加4個位元組。作為析取匹配的EFP的封裝MTU開銷被視為其最高元素的MTU。
- `encapsulation dot1q 10-100, untagged`將計為一個標籤，因為10-100範圍是最高元素。

## 非EVC(XR 12000和CRS)

Cisco XR 12000系列路由器和電信級路由系統(CRS)等路由器在子介面上使用VLAN匹配的傳統配置。這些特性適用於CRS上的L2VPN L2介面和不遵循EVC模型的XR 12000路由器：

- 在非EVC平台上，當傳入的dot1q或dot1ad標籤在L2傳輸子介面上接收到時，這些標籤將被自動剝離。
- 當您計算要發出訊號的MPLS LDP的負載大小時，從子介面的MTU中減去標籤的大小，如**show interface**命令所示。
- 這與路由的子介面的情況類似。
- 子介面從主介面繼承MTU；將每個標籤的4個位元組新增到主介面的MTU中，以便計運算元介面的MTU。例如，如果QinQ子介面有2個dot1q標籤，則預設情況下，該子介面的MTU比主介面的MTU大8個位元組。
- 您也可以可以在子介面下使用**mtu**命令，但它僅用於減少子介面的MTU（從主介面的MTU繼承）。

下面幾個示例說明了這些特性。

此示例說明如何配置非EVC子介面：

```
RP/0/RP0/CPU0:router1#sh run int gigabitEthernet 0/0/0/2.201
interface GigabitEthernet0/0/0/2.201 l2transport
dot1q vlan 201
!
```

```
RP/0/RP0/CPU0:router1#
```

非EVC平台使用**dot1q vlan**或**dot1ad vlan**命令，而不是EVC平台(ASR9000)的**encapsulation**和**rewrite**命令。

如果沒有在主介面或子介面上明確設定MTU，則預設可能會接收1500位元組的L3封包：

```
RP/0/RP0/CPU0:router1#sh int gig 0/0/0/2 | i MTU
MTU 1514 bytes, BW 1000000 Kbit (Max: 1000000 Kbit)
RP/0/RP0/CPU0:router1#sh int gig 0/0/0/2.201 | i MTU
MTU 1518 bytes, BW 1000000 Kbit (Max: 1000000 Kbit)
RP/0/RP0/CPU0:router1#
```

子介面MTU從主介面MTU(1514)計算；為每個dot1q標籤新增4個位元組。由於使用**dot1q vlan 201**命令在子介面上配置了一個標籤，因此1514中為1518位元組的MTU增加4個位元組。

MPLS LDP中對應的負載MTU為1500位元組，因為乙太網報頭的14位元組不計數，並且非EVC平台在它通過PW時自動彈出1 dot1q標籤：

```
RP/0/RP0/CPU0:router1#sh l2vpn xconnect detail
```

```
Group mtu, XC mtu, state is down; Interworking none  
AC: GigabitEthernet0/0/0/2.201, state is up  
Type VLAN; Num Ranges: 1  
VLAN ranges: [201, 201]  
MTU 1500; XC ID 0x1080001; interworking none
```

如果將主介面的MTU增加到2014位元組，則子介面的MTU會相應地增加：

```
RP/0/RP0/CPU0:router1#sh run int gig 0/0/0/2  
interface GigabitEthernet0/0/0/2  
description static lab connection to head 4/0/0 - dont change  
cdp  
mtu 2014  
ipv4 address 10.0.100.1 255.255.255.252  
load-interval 30  
!
```

```
RP/0/RP0/CPU0:router1#sh run int gig 0/0/0/2.201  
interface GigabitEthernet0/0/0/2.201 l2transport  
dot1q vlan 201  
!
```

```
RP/0/RP0/CPU0:router1#sh int gig 0/0/0/2 | i MTU  
MTU 2014 bytes, BW 1000000 Kbit (Max: 1000000 Kbit)  
RP/0/RP0/CPU0:router1#sh int gig 0/0/0/2.201 | i MTU  
MTU 2018 bytes, BW 1000000 Kbit (Max: 1000000 Kbit)  
RP/0/RP0/CPU0:router1#sh l2vpn xconnect detail
```

```
Group mtu, XC mtu, state is down; Interworking none  
AC: GigabitEthernet0/0/0/2.201, state is up  
Type VLAN; Num Ranges: 1  
VLAN ranges: [201, 201]  
MTU 2000; XC ID 0x1080001; interworking none
```

因此，為了計算MPLS LDP MTU，請減去14位元組的乙太網報頭，並為子介面下配置的每個標籤新增4個位元組。

## 自動乙太網介面驅動程式MTU和MRU配置

在乙太網介面上，介面驅動程式配置有基於介面MTU配置的MTU和MRU。

使用**show controller <interface> all**指令可檢視乙太網路介面驅動程式上已設定的MTU和MRU。

在低於Cisco IOS XR 5.1.1的版本中，乙太網介面驅動程式上的MTU和MRU是根據介面上的Cisco IOS XR MTU配置自動配置的。

在乙太網驅動程式上配置的MTU/MRU只是基於為新增2個乙太網標籤和CRC欄位而配置的MTU + 12個位元組。12位元組新增到乙太網路驅動程式MTU/MRU中，無論子介面上是否設定任何VLAN標籤。

以下範例顯示所有Cisco IOS XR版本低於Cisco IOS XR 5.1.1且在ASR 9000介面上預設的MTU為1514：

```
RP/0/RSP0/CPU0:ASR2#show interface Gi0/2/0/0  
GigabitEthernet0/2/0/0 is up, line protocol is up
```

```
Interface state transitions: 3
Hardware is GigabitEthernet, address is 18ef.63e2.0598 (bia 18ef.63e2.0598)
Description: Static_Connections_to_ME3400-1_Gi_0_2 - Do Not Change
Internet address is Unknown
MTU 1514 bytes, BW 1000000 Kbit (Max: 1000000 Kbit)
<snip>
```

MTU/MRU programmed on ethernet interface driver is 1514 + 12 bytes

```
RP/0/RSP0/CPU0:ASR2#show controllers Gi0/2/0/0 all
```

<snip>

Operational values:

```
Speed: 1Gbps
Duplex: Full Duplex
Flowcontrol: None
Loopback: None (or external)
MTU: 1526
MRU: 1526
Inter-packet gap: standard (12)
```

<snip>

在Cisco IOS XR 5.1.1及更高版本中，乙太網介面驅動程式中使用的MTU和MRU已更改，現在取決於任何子介面上配置的VLAN標籤數。

如果在任何子介面上未配置VLAN標籤，則驅動程式MTU/MRU等於介面上配置的MTU + 4 CRC位元組，例如1514 + 4 = 1518位元組。

如果在任何子介面上配置了一個VLAN，則驅動程式MTU/MRU等於配置的MTU + 8位元組（1標籤+ CRC），例如1514 + 8 = 1522位元組。

如果在任何子介面上配置了兩個VLAN標籤，則驅動程式MTU/MRU等於配置的MTU + 12位元組（2標籤+ CRC），例如1514 + 12 = 1526位元組

如果為second-do1q標籤配置帶有any關鍵字的QinQ，則驅動程式MTU/MRU等於配置的MTU + 8位元組（1標籤+ CRC），例如1514 + 8 = 1522位元組。

以下範例顯示ASR 9000上Cisco IOS XR 5.1.1版及更高版本中的行為：

```
RP/0/RSP0/CPU0:ASR2#sh run int ten0/1/0/0
interface TenGigE0/1/0/0
 cdp

RP/0/RSP0/CPU0:ASR2#show controllers ten0/1/0/0 all
```

<snip>

Operational values:

```
Speed: 10Gbps
Duplex: Full Duplex
Flowcontrol: None
Loopback: Internal
MTU: 1518
MRU: 1518
Inter-packet gap: standard (12)
```

<snip>

```
RP/0/RSP0/CPU0:ASR2#config
RP/0/RSP0/CPU0:ASR2(config-if)#int ten0/1/0/0.1
RP/0/RSP0/CPU0:ASR2(config-subif)#encapsulation dot1q 1
```

```
RP/0/RSP0/CPU0:ASR2(config-subif)#commit

RP/0/RSP0/CPU0:ASR2#show controllers ten0/1/0/0 all

<snip>
Operational values:
  Speed: 10Gbps
  Duplex: Full Duplex
  Flowcontrol: None
  Loopback: Internal
  MTU: 1522
  MRU: 1522
  Inter-packet gap: standard (12)
<snip>

RP/0/RSP0/CPU0:ASR2#config
RP/0/RSP0/CPU0:ASR2(config)#int ten0/1/0/0.2
RP/0/RSP0/CPU0:ASR2(config-subif)#encapsulation dot1q 10 second-dot1q 20
RP/0/RSP0/CPU0:ASR2(config-subif)#commit
```

```
RP/0/RSP0/CPU0:ASR2#show controllers ten0/1/0/0 all
```

```
<snip>
Operational values:
  Speed: 10Gbps
  Duplex: Full Duplex
  Flowcontrol: None
  Loopback: Internal
  MTU: 1526
  MRU: 1526
  Inter-packet gap: standard (12)
<snip>
```

```
RP/0/RSP0/CPU0:ASR2#config
RP/0/RSP0/CPU0:ASR2(config)#int ten0/2/0/0
RP/0/RSP0/CPU0:ASR2(config)#cdp
RP/0/RSP0/CPU0:ASR2(config)#int ten0/2/0/0.1 l2transport
RP/0/RSP0/CPU0:ASR2(config-subif)#encapsulation dot1q 10 second-dot1q any
RP/0/RSP0/CPU0:ASR2(config-subif)#commit
```

```
RP/0/RSP0/CPU0:ASR2#show controllers ten0/1/0/0 all
```

```
<snip>
Operational values:
  Speed: 10Gbps
  Duplex: Full Duplex
  Flowcontrol: None
  Loopback: Internal
  MTU: 1522
  MRU: 1522
  Inter-packet gap: standard (12)
<snip>
```

在大多數情況下，5.1.1版及更高版本中的行為更改不應要求對介面上的MTU配置進行任何更改。

如果子介面配置了單個VLAN標籤，則此行為更改會導致問題，但會收到具有兩個VLAN標籤的資料包。在這種情況下，收到的資料包可能會超過乙太網介面驅動程式上的MRU。為了消除這種情況，介面MTU可以增加4個位元組，或者子介面配置有兩個VLAN標籤。

5.1.1版中的自動乙太網介面驅動程式MTU和MRU配置對於CRS和ASR 9000路由器而言是相同的。但是，運行5.1.1版的CRS路由器在show controller輸出中顯示的MTU和MRU值中不包括4位元組CRC。在CRS和ASR9000之間，報告的行為不同。

```
RP/0/RP0/CPU0:CRS#sh run int ten0/4/0/0
```

```
Mon May 19 08:49:26.109 UTC
```

```
interface TenGigE0/4/0/0
```

```
<snip>
```

```
Operational values:
```

```
Speed: 10Gbps  
Duplex: Full Duplex  
Flowcontrol: None  
Loopback: None (or external)  
MTU: 1514  
MRU: 1514  
Inter-packet gap: standard (12)
```

```
RP/0/RP0/CPU0:CRS(config)#int ten0/4/0/0.1
```

```
RP/0/RP0/CPU0:CRS(config-subif)#encapsulation dot1q 1
```

```
RP/0/RP0/CPU0:CRS(config-subif)#commit
```

```
Operational values:
```

```
Speed: 10Gbps  
Duplex: Full Duplex  
Flowcontrol: None  
Loopback: None (or external)  
MTU: 1518  
MRU: 1518  
Inter-packet gap: standard (12)
```

ASR 9000上show controller輸出中顯示MTU和MRU的方式將在將來發生更改，因此CRC的4位元組不會包含在顯示的MTU/MRU值中。可使用思科錯誤ID [CSCuo93379](#)追蹤此未來的變更。

## 從低於5.1.1的版本升級到5.1.1或更高版本時，轉換配置

- 預設MTU:

如果低於5.1.1版的版本中有主介面沒有子介面，且沒有任何mtu指令:

```
interface TenGigE0/1/0/19  
l2transport  
!  
!
```

並且此介面傳輸dot1q或QinQ幀，那麼MTU在5.1.1及更新版本中應手動配置為「mtu 1522」：

```
interface TenGigE0/1/0/19  
mtu 1522  
l2transport  
!  
!
```

此配置允許QinQ幀像早期版本那樣傳輸。如果只傳輸dot1q而不傳輸QinQ，則MTU值可配置為1518。

如果存在為dot1q或QinQ配置的子介面，但使用「any」關鍵字，並且未在5.1.1之前的版本中配置具有2個顯式標籤的QinQ子介面：

```
interface TenGigE0/1/0/19
```

```
!  
interface TenGigE0/1/0/19.100 l2transport  
encapsulation dot1q 100  
!  
interface TenGigE0/1/0/19.101 l2transport  
encapsulation dot1q 101 second-dot1q any  
!
```

5.1.1及更高版本中的此配置僅允許傳輸帶有一個標籤的幀，因此，如果要傳輸QinQ幀，也應手動增加4位元組：

```
interface TenGigE0/1/0/19  
mtu 1518  
!  
interface TenGigE0/1/0/19.100 l2transport  
encapsulation dot1q 100  
!  
interface TenGigE0/1/0/19.101 l2transport  
encapsulation dot1q 101 second-dot1q any  
!
```

如果設定了2個明確標籤（不使用「any」關鍵字）的QinQ子介面，則升級到5.1.1版及更新版本時，不需要修改MTU組態：

```
interface TenGigE0/1/0/19  
!  
interface TenGigE0/1/0/19.101 l2transport  
encapsulation dot1q 101 second-dot1q 200  
!
```

如果沒有第2層傳輸子介面，但只有第3層路由介面，則預計兩端的MTU配置將匹配，並且不會出現大於所傳輸MTU的幀。升級到5.1.1版及更新版本時，無需更新MTU組態。

- 版本5.1.1之前版本中的非預設MTU:

同樣，在早於5.1.1版的版本中設定了非預設MTU，且未設定任何子介面，且必須傳輸dot1q或QinQ訊框時，在升級到5.1.1版或更新版本時，設定的MTU值應該增加8位元組。

低於版本5.1.1的版本：

```
interface TenGigE0/1/0/19  
mtu 2000  
l2transport  
!  
!
```

升級到5.1.1版及更新版本時，MTU應手動增加8位元組：

```
interface TenGigE0/1/0/19  
mtu 2008  
l2transport  
!  
!
```

如果存在dot1q子介面，並且沒有帶有second-dot1q標籤的關鍵字的QinQ子介面或QinQ子介面，則配置的MTU值也應增加4位元組。

低於版本5.1.1的版本：



```
interface TenGigE0/1/0/19
mtu 2000
!
interface TenGigE0/1/0/19.100 l2transport
encapsulation dot1q 100
!
interface TenGigE0/1/0/19.101 l2transport
encapsulation dot1q 101 second-dot1q any
!
```

版本5.1.1及更高版本：

```
interface TenGigE0/1/0/19
mtu 2004
!
interface TenGigE0/1/0/19.100 l2transport
encapsulation dot1q 100
!
interface TenGigE0/1/0/19.101 l2transport
encapsulation dot1q 101 second-dot1q any
!
```

如果設定了2個明確標籤 ( 不使用「any」關鍵字 ) 的QinQ子介面，則升級到5.1.1版及更新版本時，不需要修改MTU組態。

```
interface TenGigE0/1/0/19
!
interface TenGigE0/1/0/19.101 l2transport
encapsulation dot1q 101 second-dot1q 200
!
```

如果沒有L2傳輸子介面，而只有L3路由介面，則預計兩端的MTU配置將匹配，並且不會出現大於傳輸的MTU的幀。升級到5.1.1版及更新版本時，無需更新MTU組態。

## 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）。