

在Cisco IOS XR和Cisco IOS路由器的MTU工作情况

Contents

[Introduction](#)

[背景信息](#)

[Configure](#)

[Cisco IOS和Cisco IOS XR软件比较](#)

[路由的L3接口](#)

[默认MTU](#)

[非默认MTU](#)

[路由的L3 Sub-interface](#)

[L2VPN L2接口](#)

[EVC \(ASR9000\)](#)

[非EVC \(XR 12000和CRS\)](#)

[自动以太网接口驱动程序MTU和MRU配置](#)

[当您从版本早于版本5.1.1到版本5.1.1升级或以后时，请转换配置](#)

Introduction

本文描述在Cisco IOS XR路由器的最大传输单元(MTU)工作情况并且与Cisco IOS路由器比较那些工作情况。使用以太网虚拟连接的它也讨论在路由的第3层(L3)接口的MTU和第2层VPN (L2VPN) L2接口(EVC)和非EVC型号。本文也描述对以太网接口驱动程序MTU和最大接收单元(MRU)如何的重要更改在版本5.1.1自动地被配置及以后。

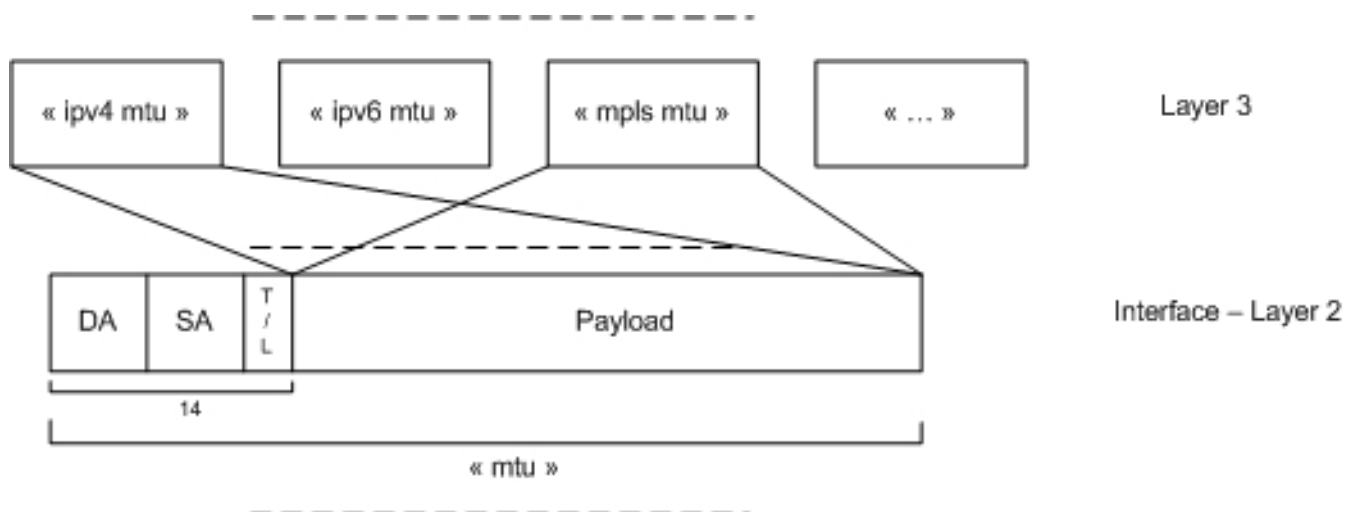
背景信息

在计算机网络，层的通信协议的MTU定义了大小，在字节层允许传输一个接口的，最大的协议数据单元。一个MTU参数与每个接口、层和协议产生关联。

在Cisco IOS XR软件的MTU特性是：

- MTU设置和显示命令，在L2和L3，包括他们的对应的层的报头大小。例如，**mtu命令**配置L2 MTU包含以太网接口的14个字节(没有dot1q)，或者点对点协议(PPP)或高级数据链路控制(HDLC)的4个字节。**mtu命令**的**ipv4**包含20字节的IPv4报头。
- 一个更高层的MTU必须在下层的有效载荷内合适。例如，如果non-dot1q以太网接口的接口MTU是1514个字节默认值，然后更高层协议例如多协议标签交换(MPLS)能有1500个字节最大值MTU在该接口的。这意味着您能适合仅一个1500个字节MPLS帧(包括标签)在以太网帧里面。如果要允许两个MPLS标记在一个1500个字节IPv4信息包顶部，您不能配置1508字节在该接

口的MPLS MTU。为了传输在以太网接口的一个1508个字节MPLS帧，必须增加接口MTU到1522或者更高的值，为了保证L2接口有效载荷是足够大运载MPLS帧。



- 在古典Cisco IOS软件(不是Cisco IOS XR软件)， **mtu命令**的接口配置L2有效载荷大小，但是不包括L2报头。在**mtu命令**的接口包括L2和L3开销的这是与Cisco IOS XR软件不同。L3 MTU发出命令，如一旦**mtu命令**的**ipv4**，配置包括L3报头该协议的最大信息包大小。这类似于Cisco IOS XR软件事例。

- 在Cisco IOS XR软件的默认接口MTU必须允许一个1500个字节L3信息包的传输。所以，默认MTU是一个主要以太网接口的1514个字节和serial interfaces的1504个字节。

本文档的剩余部分说明MTU特性，比较Cisco IOS和Cisco IOS XR软件工作情况，并且提供接口的这些类型的示例：

- 路由的L3接口
- 路由的L3 sub-interface
- L2VPN L2接口

Configure

Note:使用[命令查找工具](#) ([仅限注册用户](#)) 可获取有关本部分所使用命令的详细信息。

Note:[命令输出解释程序工具](#) ([仅限注册用户](#)) 支持某些 **show** 命令。使用输出解释器工具来查看 **show** 命令输出的分析。

Cisco IOS和Cisco IOS XR软件比较

此部分比较Cisco IOS和Cisco IOS XR软件工作情况关于MTU特性。

在Cisco IOS软件， **mtu命令**和对应**显示命令**不包括L2报头。请使用**mtu命令**为了配置L2有效载荷到L3信息包的最大大小，包括L3报头。

这是与Cisco IOS XR软件不同， **mtu命令**包括L2报头(以太网的14个字节或PPP/HDLC的4个字节)。

如果Cisco IOS路由器配置有mtu x和被连接到Cisco IOS XR路由器，则应该配置有在Cisco IOS XR路由器的对应接口以太网接口的serial interfaces的mtu x+14或者mtu x+4。

Cisco IOS和Cisco IOS XR软件有ipv4 mtu，IPv6 mtu和mpls mtu命令的同一含义;必须配置有他们同样值。

结果，这是在Cisco IOS软件的配置在以太网接口：

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

在Cisco IOS XR软件相邻的对应的配置是：

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

路由的L3接口

MTU值必须是相同的在所有设备被连接到L2网络。否则，这些症状也许报告：

- 中间系统对中间系统(IS-IS)邻接不出来。默认情况下，IS-IS用途hello-padding;因此，hello也许被分析作为巨人，并且也许丢弃，当一个路由器有比值低时的MTU值在其他路由器。
- 因为大数据包描述符(DBD)信息包也许被分析作为巨人，并且也许丢弃，开放最短路径优先(OSPF)邻接陷在Exstart或交换状态。当信息包在有更小的MTU值的时一个路由器收到，数据库没有同步。
- 数据流量被分析作为巨人和降低，当在有比那个低时的MTU值的一个设备被接受在发送设备。
- 当大信息包被撤销时，有低吞吐量。在路径MTU发现的情况下，TCP会话能恢复，当大信息包被丢弃时，但是这影响吞吐量。

默认MTU

当没有配置时，此部分分析路由接口的默认MTU mtu命令：

```
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: 0x000000000010059f (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#
```

在本例中，默认L2接口MTU是1514个字节并且包含14字节的以太网报头。14个字节由6字节的目的地MAC地址，6字节的源MAC地址和2字节的类型或长度组成。这包括前导、帧分隔符、4字节的帧检查顺序(FCS)和帧间的空白。对于PPP或HDLC帧，4字节的L2报头占；如此默认接口MTU是1504个字节。

L3儿童协议继承他们的从父母MTU的有效载荷的MTU。当您从L2 1514个字节时MTU减去14字节的L2报头，您有1500个字节L2有效载荷。这成为L3协议的MTU。IPv4、IPv6、MPLS和无连接网络服务(CLNS)继承this1500字节MTU。默认情况下，结果，Cisco IOS XR以太网接口能传输是相同的作为在Cisco IOS以太网接口的default的一个1500个字节L3信息包。

非默认MTU

此部分显示如何配置mpls mtu 1508为了发送1500个字节IPv4信息包与4个字节两个MPLS标记的中的每一个，在信息包顶部：

```
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#

虽然**mpls mtu 1508**命令被付诸，不适用，因为MPLS仍然有1500个字节MTU在**show**命令。这是因为L3儿童协议大于他们的父母L2接口有效载荷不能有MTU。

为了允许两个标签在1500字节IP信息包顶部，您必须：

- 配置1522个字节L2接口MTU，因此所有儿童协议(包括MPLS)继承MTU 1508个字节(1522 - 14 = 1508)。
- 使L3协议的MTU降低到1500个字节，因此仅MPLS允许超出1500个字节。

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#
```

此配置让您发送1500个字节IPv4和IPv6信息包和MPLS信息包1508个字节(有两个标记的一个1500字节信息包在上面)。

路由的L3 Sub-interface

这些特性适用路由的L3 sub-interface。

一路路由的sub-interface MTU继承其父母主要接口MTU;添加在sub-interface配置的每个VLAN标记的4个字节。因此，有dot1q sub-interface的4个字节和建立隧道(QinQ) sub-interface的IEEE 802.1Q的8个字节。

结果，L3相同大小的信息包在主要接口和sub-interface可以转发。

mtu命令可以被配置在sub-interface下，但是适用，只有当与从主要接口被继承的MTU是低或相等的。

这是主要接口MTU是2000个字节的示例：

```
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#
```

在显示命令， sub-interface的MTU是2004年;因为有一个dot1q标记被配置在sub-interface下，请添加4个字节到主要接口的MTU。

然而， IPv4和IPv6信息包的MTU仍然是相同的象那主要接口(1986)。这是因为L3协议的MTU当前被计算如下：2004年- 14 - 4 = 1986年。

mtu命令可以被配置在sub-interface下，但是适用被配置的MTU，只有当与从主要接口的MTU是低或相等的(4个字节被继承大于主要接口的MTU)。

当大于被继承的MTU sub-interface的MTU，它不适用，如显示这里：

```
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，您能使用只**mtu命令**为了降低从主要接口继承的MTU值。

同样地，您能也使用L3协议(IPv4， IPv6 MTU命令， MPLS)为了降低从sub-interface L2有效载荷MTU的继承的值L3。L3协议MTU不生效，当被配置对不适合L2 MTU的有效载荷的值时。

L2VPN L2接口

L2VPN的MTU是重要的，因为标签转发协议(LDP)不带来一pseudowire (PW)，当在附件电路的MTU在PW的每边不是相同的时。

这是**show命令**说明L2VPN PW坚持下来，当有MTU不匹配时：

```
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#

```

在本例中，请注意MPLS L2VPN运营商边缘(观点扫描器)在每边必须发信号同一MTU值为了提出PW。

MPLS发信号的MTU LDP不包括L2开销。这是与XR接口设置不同并且显示包括L2开销的命令。在sub-interface的MTU是2018个字节(如被继承从2014个字节主要接口)，但是LDP发了信号2000个字节MTU。结果，它从L2报头减去18个字节(14字节的以太网报头+ 4字节的1个dot1q标记)。

知道是重要的每个设备如何计算附件电路的MTU值为了改正MTU不匹配。这取决于参数例如供应商、平台、软件版本和配置。

EVC (ASR9000)

Cisco ASR 9000系列聚合服务路由器使用EVC基础设施型号，允许配比在L2VPN L2接口和sub-interface的灵活的VLAN。

EVC L2VPN L2接口有这些特性：

- 他们允许一个或更多标记的配置用**encapsulation**命令。
- 默认情况下和用**encapsulation**命令，标记在PWs保留并且被传输。默认情况下结果，当您在非EVC平台，需要执行您不需要剥离标记。
- 当您决定弹出流入标记或推进一些另外的标记在流入的帧顶部时，请使用**重写**命令。

为了计算sub-interface MTU，取出主要接口MTU (默认值或那个手工被配置在主要接口下)和添加用**encapsulation**命令配置的每个VLAN标记的4个字节。请参阅[特定EFP封装命令](#)。

当**mtu**命令在sub-interface下时，生效，只有当低于计算MTU。**重写**命令不影响sub-interface的MTU。

示例如下：

```
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，在主要接口的MTU是2014个字节;因为有两个标记被配置在sub-interface下，请添加8个字节。

如果配置**mtu 2026**字节在sub-interface下，不适用，因为大于从主要接口继承的sub-interface的MTU (2022)。结果，您能配置仅sub-interface MTU更低比2022个字节。

基于此sub-interface MTU，请计算发信号给相邻MPLS LDP有效载荷的MTU，并且切记是相同的到远程L2VPN PE计算的那个。这是**重写**命令开始活动的地方。

为了计算MPLS LDP有效载荷的MTU，请采取sub-interface的MTU，则：

1. 减去以太网报头的14个字节。
2. 减去在**重写**命令弹出的每个标记的4个字节被配置在sub-interface下。
3. 添加在**重写**命令推进的每个标记的4个字节被配置在sub-interface下。

这是与QinQ配置的同个示例在gig 0/1/0/3.201 :

```
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)
```

这是MPLS LDP有效载荷的MTU的计算 :

1. sub-interface MTU的MTU值 : 2022个字节
2. 减去14字节的以太网报头 : $2022 - 14 = 2008$ 个字节
3. 减去每个弹出的标记的4个字节在**重写**命令 : $2008 - 4 * 2 = 2000$ 年

保证远端通告2000个字节MPLS LDP有效载荷。否则 , 请调整本地或远程附件电路(AC) MTU大小他们如此匹配。

```
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
```

特定以太网流点(EFP)封装命令

这些封装算作是配比的零的标记 , 因此他们不增加sub-interface MTU :

- 无示踪的封装
- encapsulation default

这些封装修正值不影响要求的标记的数量为了计算sub-interface MTU :

- 本地
- 有效载荷以太网类型
- 确切
- COS
- 入口源MAC或入口目的地MAC

封装[dot1q]dot1ad]优先级标记为的算作是匹配单个标记。

当最内在的标记匹配使用的‘任何’关键字不增加sub-interface MTU。

- **encapsulation dot1q**其中任一不增加sub-interface MTU。
- **封装dot1ad 10 dot1q**其中任一认为，一个标记;它增加sub-interface MTU 4个字节。
- **封装dot1ad所有dot1q 7**认为作为两个标记;它增加sub-interface MTU 8个字节。

VLAN ID的范围增加sub-interface MTU：

• **encapsulation dot1q 10-100**认为，一个标记;它增加sub-interface MTU 4个字节。
是分离匹配EFP的封装MTU开销对待其最高的元素MTU。

- 因为范围10 -100是最高的元素， **encapsulation dot1q 10-100**，无示踪认为，因为一个标记。

非EVC (XR 12000和CRS)

路由器类似Cisco XR 12000系列路由器和载波路由系统(CRS)使用传统配置配比在sub-interface的VLAN。这些特性应用于L2VPN L2接口在CRS和在XR不跟随EVC型号的12000路由器：

- 在非EVC平台上，流入dot1q或dot1ad标记，当他们在L2传输sub-interface时，被接受自动地剥离。
- 当您计算MPLS的LDP有效载荷大小发信号时，从sub-interface的MTU请减去标记的大小，如在**show interface命令中看到**。
- 这类似于一路由的sub-interface的事例。
- sub-interface继承其从主要接口的MTU;添加每个标记的4个字节到主要接口的MTU为了计算sub-interface的MTU。例如，如果QinQ sub-interface有2个dot1q标记，默认情况下，sub-interface有大于主要接口的MTU是8个的字节MTU。
- 您能也使用**mtu命令**在sub-interface下，但是用于只减少sub-interface的MTU，从主要接口的MTU被继承。

这是说明这些特性的几个示例。

此示例显示如何配置一非EVC sub-interface：

```
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**封装**并且**重写**EVC平台(ASR9000)的命令。

如果在主要或sub-interface不明确配置MTU，默认情况下则一个1500个字节L3信息包可以收到：

```
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#
```

sub-interface MTU从主要接口MTU (1514)被计算;添加每个dot1q标记的4个字节。由于有在sub-interface配置的一个标记用**dot1q vlan 201**命令，请添加4个字节到1514 1518个字节MTU的。

在MPLS LDP的对应的有效载荷MTU是1500个字节，因为14字节的以太网报头没有被计数，并且这

一个dot1q标记乘非EVC平台自动地弹出，当在PW时去：

```
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
```

如果增加主要接口的MTU对2014个字节，相应地增加sub-interface的MTU：

```
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，为了计算MPLS LDP MTU，减去14字节的以太网报头和添加每个标记的4个字节被配置在sub-interface下。

自动以太网接口驱动程序MTU和MRU配置

在以太网接口接口驱动程序配置有根据接口MTU配置的MTU和MRU。

被配置的MTU和MRU在以太网接口驱动程序能在all命令show controller的<interface>看到。

在版本中早于Cisco IOS XR Release 5.1.1，MTU和MRU在以太网接口驱动程序根据Cisco IOS XR MTU配置自动地被配置了接口。

在以太网驱动程序配置的MTU/MRU根据被配置的MTU+ 2个以太网标记和CRC字段的添加的12个字节。如果有在sub-interface，配置的任何VLAN标记12个字节被添加了到以太网驱动程序MTU/MRU不管。

与所有Cisco IOS XR版本的一个示例早于Cisco IOS XR Release 5.1.1和默认MTU 1514在ASR 9000接口显示得这里：

```
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 Release 5.1.1和以后，在以太网接口驱动程序使用的MTU和MRU根据的VLAN标记的数量在被配置任何sub-interface更改了和当前。

如果VLAN标记在任何sub-interface没有被配置，驱动程序MTU/MRU等于在接口的被配置的MTU + 4个CRC字节，例如1514个+ 4个= 1518个字节。

如果一个VLAN在任何sub-interface被配置，驱动程序MTU/MRU等于被配置的MTU+ 8个字节(1个标记+ CRC)，例如1514个+ 8个= 1522个字节。

如果两个VLAN标记在任何sub-interface被配置，驱动程序MTU/MRU等于被配置的MTU+ 12个字节(2个标记+ CRC)，例如1514个+ 12个= 1526个字节

如果与任何关键字的QinQ为second-do1q标记被配置驱动程序MTU/MRU等于被配置的MTU+ 8个字节(1个标记+ CRC)，例如1514个+ 8个= 1522个字节。

这些示例显示在Cisco IOS XR Release 5.1.1和稍后ASR 9000的工作情况：

```
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>
```

在大多数情况下在版本5.1.1上的此工作情况变化及以后不应该要求对MTU的配置的任何更改在接口的。

此工作情况更改能引起问题一旦sub-interface配置有单个VLAN标记，但是收到与两个VLAN标记的信息包。在该情况下收到的可以超出在以太网接口驱动程序的信息包MRU。为了排除该情况，接口MTU可能按4个字节或sub-interface增加配置有两个VLAN标记。

自动以太网接口驱动程序MTU和在版本5.1.1工作情况的MRU配置是相同的为CRS和ASR 9000路由器。但是运行版本5.1.1的CRS路由器不包含4字节CRC按在**show controller**输出中显示的MTU和MRU值。工作情况如何报告不是相同的在CRS和ASR9000之间。

```
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)
```

方式MTU和MRU在ASR 9000的show controller输出中显示在将来将更改，以便4字节的CRC不会包含按MTU/MRU值显示了。此将来更改可以跟踪与Cisco Bug ID [CSCuo93379](#)。

当您从版本早于版本5.1.1到版本5.1.1升级或以后时，请转换配置

- 默认MTU :

如果早于版本5.1.1有一个主要接口没有任何sub-interface和没有其中任一**mtu**命令在版本 :

```
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)
```

并且此接口传输dot1q或QinQ帧，然后应该手工配置MTU到“在版本5.1.1的mtu 1522"及以后：

```
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)
```

此配置在更早版本允许QinQ帧被传输正如。如果而不是仅dot1q QinQ将被传输，MTU值可能被配置到1518。

如果有sub-interface配置有为dot1q或QinQ，但是“任何”关键字和QinQ sub-interface与2个明确标记在版本早于版本5.1.1未被配置：

```
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)
```

在版本5.1.1的此配置及以后只将允许到有一个标记的传输帧，因此应该被4个字节手工也增加MTU，如果QinQ帧将被传输：

```
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)
```

如果不使用“任何”关键字的2个明确标记(配置与的一QinQ sub-interface，没有需要修改MTU配置，当您升级到版本5.1.1及以后时：

```
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)
```

如果没有L2传输仅sub-interface，但是L3路由接口，预计MTU配置配比在两边，并且大于被传输的MTU没有的帧。当您升级到版本5.1.1及以后时，没有需要更新MTU配置。

- 在版本的非默认MTU早于版本5.1.1：

同样地，当非默认MTU在的版本早于未配置被配置了版本5.1.1和sub-interface，并且dot1q或QinQ帧必须被传输，然后应该按8个字节增加被配置的MTU值，当您升级到版本5.1.1或以上时。

早于版本5.1.1发布：

```
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)
```

当您升级到版本5.1.1及以后时，应该按8个字节手工增加MTU：

```
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)
```

如果有一dot1q sub-interface和没有QinQ sub-interface或者一QinQ sub-interface与任何关键字second-dot1q标记的，应该按4个字节也增加被配置的MTU值。

早于版本5.1.1发布：

```
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)
```

版本5.1.1及以后：

```
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)
```

如果不使用“任何”关键字的2个明确标记(配置与的一QinQ sub-interface，没有需要修改MTU配置，当您升级到版本5.1.1及以后时。

```
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)
```

如果没有L2仅传输sub-interface，但是L3路由接口，预计MTU配置配比在两边，并且大于被传输的MTU没有的帧。当您升级到版本5.1.1及以后时，没有需要更新MTU配置。