

Comportamiento de MTU en Cisco IOS XR y routers Cisco IOS

Contenido

[Introducción](#)

[Antecedentes](#)

[Configurar](#)

[Comparación de Cisco IOS y Cisco IOS XR Software](#)

[Interfaces L3 enrutadas](#)

[MTU predeterminada](#)

[MTU no predeterminada](#)

[Subinterfaces L3 enrutadas](#)

[Interfaz L2VPN L2](#)

[EVC \(ASR9000\)](#)

[Sin EVC \(XR 12000 y CRS\)](#)

[Configuración automática de MTU y MRU del controlador de interfaz Ethernet](#)

[Convierta la configuración cuando actualice de una versión anterior a la versión 5.1.1 a la versión 5.1.1 o posterior](#)

Introducción

Este documento describe los comportamientos de la unidad de transmisión máxima (MTU) en los routers Cisco IOS[®] XR y compara esos comportamientos con los routers Cisco IOS. También analiza las MTU en las interfaces enrutadas de capa 3 (L3) y las interfaces L2 de VPN de capa 2 (L2VPN) que utilizan los modelos de conexión virtual Ethernet (EVC) y no EVC. Este documento también describe cambios importantes en cómo la MTU del controlador de interfaz Ethernet y la unidad de recepción máxima (MRU) se configuran automáticamente en la versión 5.1.1 y posteriores.

Antecedentes

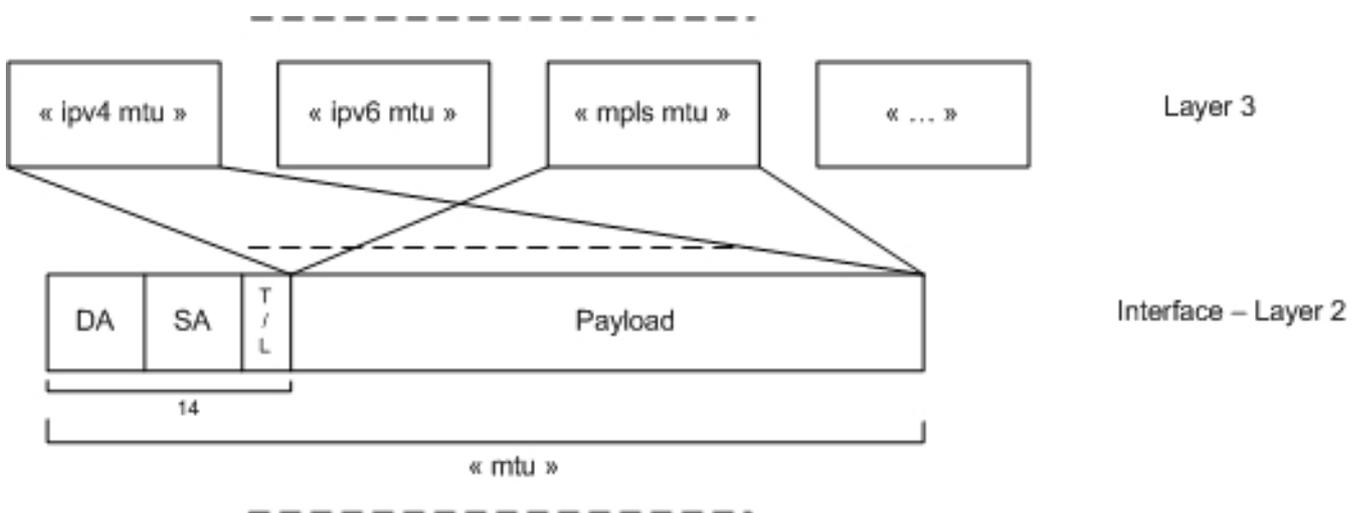
En las redes informáticas, la MTU de un protocolo de comunicaciones de una capa define el tamaño, en bytes, de la unidad de datos de protocolo más grande que la capa puede transmitir a través de una interfaz. Un parámetro de MTU se asocia a cada interfaz, capa y protocolo.

Las características de MTU en el software Cisco IOS XR son:

- Los comandos **config** y **show** de MTU, en L2 y L3, incluyen el tamaño del encabezado de su capa correspondiente. Por ejemplo, el comando **mtu** que configura la MTU de capa 2 incluye 14 bytes para una interfaz Ethernet (sin dot1q) o 4 bytes para el protocolo punto a punto

(PPP) o el control de link de datos de alto nivel (HDLC). El comando **ipv4 mtu** incluye 20 bytes del encabezado IPv4.

- La MTU de una capa superior debe caber dentro de la carga útil de la capa inferior. Por ejemplo, si la MTU de interfaz de una interfaz Ethernet sin dot1q es la predeterminada de 1514 bytes, los protocolos de capa más alta como Multiprotocol Label Switching (MPLS) pueden tener una MTU máxima de 1500 bytes en esa interfaz. Esto significa que sólo puede caber una trama MPLS de 1500 bytes (incluidas las etiquetas) dentro de la trama Ethernet. No puede configurar una MTU MPLS de 1508 bytes en esa interfaz si desea permitir dos etiquetas MPLS sobre un paquete IPv4 de 1500 bytes. Para transmitir una trama MPLS de 1508 bytes en una interfaz Ethernet, la MTU de la interfaz debe aumentarse a 1522, o un valor superior, para garantizar que la carga útil de la interfaz L2 sea lo suficientemente grande como para transportar la trama MPLS.



- En el software Cisco IOS clásico (no en el software Cisco IOS XR), el comando **interface mtu** configura el tamaño de la carga útil L2, pero no incluye el encabezado L2. Esto es diferente del software Cisco IOS XR que incluye la sobrecarga L2 y L3 en el comando **interface mtu**. Los comandos L3 MTU, como en el caso del comando **ipv4 mtu**, configuran el tamaño máximo de paquete de ese protocolo que incluye el encabezado L3. Esto es similar al caso del software Cisco IOS XR.
- La MTU de interfaz predeterminada en el software Cisco IOS XR debe permitir el transporte de un paquete L3 de 1500 bytes. Por lo tanto, la MTU predeterminada es de 1514 bytes para una interfaz Ethernet principal y de 1504 bytes para una interfaz serial.

El resto de este documento ilustra las características de MTU, compara el comportamiento del software Cisco IOS y Cisco IOS XR, y brinda ejemplos para estos tipos de interfaces:

- Interfaces L3 enrutadas
- Subinterfaces L3 enrutadas
- Interfaces L2VPN L2

Configurar

Nota: Use el Command Lookup Tool (únicamente clientes registrados) para obtener más

información sobre los comandos que se utilizan en esta sección.

Nota: la [herramienta Output Interpreter Tool](#) (sólo clientes [registrados](#)) admite ciertos comandos **show**. Utilice la herramienta para ver un análisis de información de salida del comando show.

Comparación de Cisco IOS y Cisco IOS XR Software

Esta sección compara el comportamiento del software Cisco IOS y Cisco IOS XR con referencia a las características de MTU.

En Cisco IOS Software, el comando **mtu** y los comandos **show** correspondientes no incluyen el encabezado L2. Utilice el comando **mtu** para configurar la carga útil L2 al tamaño máximo para los paquetes L3, incluido el encabezado L3.

Esto es diferente del software Cisco IOS XR, donde el comando **mtu** incluye el encabezado L2 (14 bytes para Ethernet o 4 bytes para PPP/HDLC).

Si un router Cisco IOS se configura con **mtu x** y se conecta a un router Cisco IOS XR, la interfaz correspondiente en el router Cisco IOS XR se debe configurar con **mtu x+14** para las interfaces Ethernet, o **mtu x+4** para las interfaces seriales.

Cisco IOS y el software Cisco IOS XR tienen el mismo significado para los comandos **ipv4 mtu**, **ipv6 mtu** y **mpls mtu**; deben configurarse con los mismos valores.

Como resultado, esta es la configuración en el software del IOS de Cisco en una interfaz Ethernet:

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

La configuración correspondiente en el vecino del software Cisco IOS XR es:

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

Interfaces L3 enrutadas

Los valores de MTU deben ser los mismos en todos los dispositivos conectados a una red L2. De lo contrario, se pueden notificar estos síntomas:

- Las adyacencias Sistema intermedio a sistema intermedio (IS-IS) no se activan. De forma predeterminada, IS-IS utiliza hello-padding; por lo tanto, los saludos podrían caracterizarse como gigantes y podrían descartarse cuando un router tiene un valor de MTU inferior a los valores de los otros routers.
- Las adyacencias OSPF (ruta de acceso más corta primero) se atascan en el estado Exstart o Exchange, ya que los paquetes de descriptor de base de datos (DBD) de gran tamaño

pueden caracterizarse como gigantes y descartarse. Cuando los paquetes se reciben en un router con un valor de MTU inferior, las bases de datos no se sincronizan.

- El tráfico de datos se caracteriza como gigantes y se descarta cuando se recibe en un dispositivo con un valor de MTU inferior al del dispositivo de transmisión.
- El rendimiento es bajo cuando se descartan paquetes grandes. En caso de detección de MTU de trayecto, la sesión TCP puede recuperarse cuando se descartan paquetes grandes, pero esto afecta el rendimiento.

MTU predeterminada

Esta sección analiza la MTU predeterminada de una interfaz ruteada cuando el comando `mtu` no está configurado:

```
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#
```

En este ejemplo, la MTU predeterminada de la interfaz L2 es de 1514 bytes e incluye 14 bytes de encabezado Ethernet. Los 14 bytes están representados por 6 bytes de dirección MAC de destino, 6 bytes de dirección MAC de origen y 2 bytes de tipo o longitud. Esto no incluye el preámbulo, el delimitador de trama, los 4 bytes de la secuencia de verificación de trama (FCS) y la brecha entre tramas. Para una trama PPP o HDLC, se contabilizan 4 bytes del encabezado L2; por lo tanto, la MTU de interfaz predeterminada es de 1504 bytes.

Los protocolos secundarios L3 heredan su MTU de la carga útil de la MTU principal. Cuando se restan 14 bytes de un encabezado L2 de una MTU L2 de 1514 bytes, se tiene una carga útil L2 de 1500 bytes. Esto se convierte en la MTU para los protocolos L3. IPv4, IPv6, MPLS y el servicio de red sin conexión (CLNS) heredan esta MTU de 1500 bytes. Como resultado, una interfaz Ethernet Cisco IOS XR, de forma predeterminada, puede transportar un paquete L3 de 1500 bytes que es el mismo que el valor predeterminado en una interfaz Ethernet Cisco IOS.

MTU no predeterminada

Esta sección muestra cómo configurar una **mtu mpls** de 1508 para enviar un paquete IPv4 de 1500 bytes con dos etiquetas MPLS de 4 bytes cada una, en la parte superior del paquete:

```
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: 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#
```

Aunque se confirma el comando **mpls mtu 1508**, no se aplica, porque MPLS todavía tiene una MTU de 1500 bytes en el comando **show**. Esto se debe a que los protocolos secundarios L3 no pueden tener una MTU mayor que la carga útil de su interfaz L2 principal.

Para permitir dos etiquetas en la parte superior de un paquete IP de 1500 bytes, debe:

- Configure una MTU de interfaz L2 de 1522 bytes, de modo que todos los protocolos secundarios (incluida MPLS) hereden una MTU de 1508 bytes (1522 - 14 = 1508).
- Reduzca la MTU de los protocolos L3 a 1500 bytes, de modo que solo se permita que MPLS exceda los 1500 bytes.

```
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: 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, 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#
```

Esta configuración le permite enviar paquetes IPv4 e IPv6 de 1500 bytes y paquetes MPLS de 1508 bytes (un paquete de 1500 bytes con dos etiquetas en la parte superior).

Subinterfaces L3 enrutadas

Estas características se aplican a las subinterfaces L3 enrutadas.

Una MTU de subinterfaz enrutada hereda la MTU de su interfaz principal principal principal; agregue 4 bytes para cada etiqueta VLAN configurada en la subinterfaz. Por lo tanto, hay 4 bytes para una subinterfaz dot1q y 8 bytes para una subinterfaz de tunelización IEEE 802.1Q (QinQ).

Como resultado, los paquetes L3 del mismo tamaño se pueden reenviar tanto en la interfaz principal como en la subinterfaz.

El comando `mtu` se puede configurar en la subinterfaz, pero se aplica sólo si es menor o igual a la MTU que se hereda de la interfaz principal.

Este es un ejemplo donde la MTU de la interfaz principal es de 2000 bytes:

```
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#
```

En los comandos **show**, la MTU de la subinterfaz es 2004; agregue 4 bytes a la MTU de la interfaz principal porque hay una etiqueta dot1q configurada bajo la subinterfaz.

Sin embargo, la MTU de los paquetes IPv4 e IPv6 sigue siendo la misma que la de la interfaz principal (1986). Esto se debe a que la MTU de los protocolos L3 se calcula ahora como: $2004 - 14 - 4 = 1986$.

El comando **mtu** se puede configurar en la subinterfaz, pero la MTU configurada se aplica sólo si es menor o igual a la MTU heredada de la interfaz principal (4 bytes más grande que la MTU de la interfaz principal).

Cuando la MTU de la subinterfaz que es más grande que la MTU heredada, no se aplica, como se muestra aquí:

```
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#
```

Por lo tanto, puede utilizar solamente el comando **mtu** para reducir el valor de MTU heredado de la interfaz principal.

De manera similar, también puede utilizar los comandos MTU de los protocolos L3 (IPv4, IPv6, MPLS) para reducir el valor de la MTU L3 heredada de la carga útil L2 de la subinterfaz. La MTU del protocolo L3 no tiene efecto cuando se configura en un valor que no cabe en la carga útil de la MTU L2.

Interfaz L2VPN L2

La MTU para una L2VPN es importante porque el protocolo de distribución de etiquetas (LDP) no activa un pseudowire (PW) cuando las MTU en los circuitos de conexión en cada lado de un PW no son las mismas.

Este es un comando **show** que ilustra que un PW L2VPN permanece inactivo cuando hay una discordancia de MTU:

```
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#

En este ejemplo, observe que los bordes del proveedor de MPLS L2VPN (PEs) en cada lado deben señalar el mismo valor de MTU para activar el PW.

La MTU señalada por MPLS LDP no incluye la sobrecarga de L2. Esto es diferente de los comandos XR interface **config** y **show** que incluyen la sobrecarga L2. La MTU en la subinterfaz es de 2018 bytes (como se hereda de la interfaz principal de 2014 bytes), pero el LDP señaló una MTU de 2000 bytes. Como resultado, resta 18 bytes (14 bytes de encabezado Ethernet + 4 bytes de 1 etiqueta dot1q) del encabezado L2.

Es importante entender cómo cada dispositivo calcula los valores de MTU de los circuitos de conexión para corregir las discordancias de MTU. Esto depende de parámetros como el proveedor, la plataforma, la versión del software y la configuración.

EVC (ASR9000)

El router de servicios de agregación Cisco ASR serie 9000 utiliza el modelo de infraestructura EVC, que permite la coincidencia de VLAN flexible en interfaces L2VPN L2 y subinterfaces.

Las interfaces L2VPN L2 de EVC tienen estas características:

- Permiten la configuración de una o más etiquetas con el comando **encapsulation**.
- De forma predeterminada y con sólo el comando **encapsulation**, las etiquetas se conservan y transportan a través de los PW. Como resultado, no es necesario eliminar etiquetas de forma predeterminada, como se hace en las plataformas que no son de EVC.
- Utilice el comando **rewrite** cuando decida hacer estallar las etiquetas entrantes o presionar algunas etiquetas adicionales sobre el marco entrante.

Para calcular la MTU de la subinterfaz, tome la MTU de la interfaz principal (ya sea la predeterminada o la configurada manualmente en la interfaz principal) y agregue 4 bytes para cada etiqueta VLAN configurada con el comando **encapsulation**. Consulte [Comandos específicos de encapsulación EFP](#).

Cuando hay un comando **mtu** bajo la subinterfaz, sólo tiene efecto si es menor que la MTU calculada. El comando **rewrite** no influye en la MTU de la subinterfaz.

Aquí tiene un ejemplo:

```
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)
```

En este ejemplo, la MTU en la interfaz principal es de 2014 bytes; agregue 8 bytes porque hay dos etiquetas configuradas bajo la subinterfaz.

Si configura **mtu 2026** bytes bajo la subinterfaz, no se aplica porque es más grande que la MTU de la subinterfaz heredada de la interfaz principal (2022). Como resultado, puede configurar solamente una MTU de subinterfaz inferior a 2022 bytes.

Según esta MTU de subinterfaz, calcule la MTU de la carga útil MPLS LDP que se señala al vecino y asegúrese de que sea idéntica a la calculada por el PE L2VPN remoto. Aquí es donde entra en juego el comando **rewrite**.

Para calcular la MTU de la carga útil MPLS LDP, tome la MTU de la subinterfaz y, a continuación:

1. Reste 14 bytes para el encabezado Ethernet.
2. Reste 4 bytes por cada etiqueta que aparezca en el comando **rewrite** configurado bajo la subinterfaz.
3. Agregue 4 bytes para cada etiqueta insertada en el comando **rewrite** configurado bajo la subinterfaz.

Este es el mismo ejemplo con la configuración QinQ en 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)
```

Este es el cálculo para la MTU de la carga útil MPLS LDP:

1. Valor de MTU de la MTU de la subinterfaz: 2022 bytes
2. Reste 14 bytes del encabezado Ethernet: $2022 - 14 = 2008$ bytes
3. Reste 4 bytes por cada etiqueta reventada en el **reescribir** comando: $2008 - 4 * 2 = 2000$

Asegúrese de que el lado remoto anuncie una carga útil MPLS LDP de 2000 bytes. De lo contrario, ajuste el tamaño de MTU del circuito de conexión local o remoto (AC) para que coincida.

```
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
```

Comandos de encapsulación de punto de flujo Ethernet (EFP) específico

Estas encapsulaciones cuentan como etiquetas cero coincidentes, por lo que no aumentan la MTU de la subinterfaz:

- **encapsulation untagged**
- **encapsulation default**

Estos modificadores de encapsulación no afectan el número de etiquetas necesarias para calcular la MTU de la subinterfaz:

- nativo
- payload-ethertype
- exacto
- cos
- ingress source-mac o ingress destination-mac

encapsulation [dot1q|dot1ad] priority-tagged cuenta como coincidente con una sola etiqueta.

La palabra clave 'any' utilizada como la coincidencia de etiqueta más interna no aumenta la MTU de la subinterfaz.

- **encapsulation dot1q any** no aumenta la MTU de la subinterfaz.
- **encapsulation dot1ad 10 dot1q any** se considera una etiqueta; aumenta la MTU de la subinterfaz en 4 bytes.
- **encapsulation dot1ad any dot1q 7** se contabiliza como dos etiquetas; aumenta la MTU de la subinterfaz en 8 bytes.

Los rangos de VLAN-ID incrementan la MTU de la subinterfaz:

- **encapsulation dot1q 10-100** se contabiliza como una etiqueta; aumenta la MTU de la subinterfaz en 4 bytes.

La sobrecarga de MTU de encapsulación de un EFP que es una coincidencia disyuntiva se trata como la MTU de su elemento más alto.

- **encapsulation dot1q 10-100, untagged** se contabiliza como una etiqueta porque el rango 10 - 100 es el elemento más alto.

Sin EVC (XR 12000 y CRS)

Los routers como el router de la serie Cisco XR 12000 y el Carrier Routing System (CRS) utilizan la configuración tradicional para la coincidencia de VLAN en las subinterfaces. Estas características se aplican a las interfaces L2VPN L2 en CRS y en los routers XR 12000 que no siguen el modelo EVC:

- En las plataformas que no son EVC, las etiquetas dot1q o dot1ad entrantes se eliminan automáticamente cuando se reciben en una subinterfaz de transporte L2.
- Cuando esté calculando el tamaño de carga útil para que MPLS LDP emita una señal, reste el tamaño de las etiquetas de la MTU de la subinterfaz, como se ve en el comando **show interface**.
- Esto es similar al caso de una subinterfaz ruteada.
- La subinterfaz hereda su MTU de la interfaz principal; agregue los 4 bytes para cada etiqueta a la MTU de la interfaz principal para calcular la MTU de la subinterfaz. Por ejemplo, si una subinterfaz QinQ tiene 2 etiquetas dot1q, la subinterfaz, de forma predeterminada, tiene una MTU que es 8 bytes más grande que la MTU de la interfaz principal.
- También puede utilizar el comando **mtu** en la subinterfaz, pero se utiliza solamente para reducir la MTU de la subinterfaz, que se hereda de la MTU de la interfaz principal.

A continuación se incluyen varios ejemplos que ilustran estas características.

Este ejemplo muestra cómo se configura una subinterfaz no EVC:

```
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#
```

Las plataformas que no son EVC utilizan los comandos **dot1q vlan** o **dot1ad vlan** en lugar de los comandos **encapsulation** y **rewrite** de las plataformas EVC (ASR9000).

Si no configura una MTU explícitamente en la interfaz principal o la subinterfaz, se puede recibir un paquete L3 de 1500 bytes de forma predeterminada:

```
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#
```

La MTU de la subinterfaz se calcula a partir de la MTU de la interfaz principal (1514); agregue 4 bytes para cada etiqueta dot1q. Debido a que hay una etiqueta configurada en la subinterfaz con el comando **dot1q vlan 201**, agregue 4 bytes a 1514 para una MTU de 1518 bytes.

La MTU de carga útil correspondiente en MPLS LDP es de 1500 bytes, ya que los 14 bytes del encabezado Ethernet no se cuentan y la etiqueta dot1q aparece automáticamente en la plataforma que no es EVC cuando pasa por PW:

```
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
```

Si aumenta la MTU de la interfaz principal a 2014 bytes, la MTU de la subinterfaz aumenta en consecuencia:

```
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
```

Por lo tanto, para calcular la MTU MPLS LDP, reste 14 bytes del encabezado Ethernet y agregue 4 bytes para cada etiqueta configurada bajo la subinterfaz.

Configuración automática de MTU y MRU del controlador de interfaz Ethernet

En las interfaces Ethernet, el controlador de interfaz se configura con una MTU y una MRU basadas en la configuración de la MTU de la interfaz.

La MTU y la MRU configuradas en el controlador de interfaz Ethernet se pueden ver con el comando **show controller <interface> all**.

En las versiones anteriores a Cisco IOS XR Release 5.1.1, la MTU y la MRU en el controlador de interfaz Ethernet se configuraban automáticamente en función de la configuración de MTU de Cisco IOS XR en la interfaz.

La MTU/MRU configurada en el controlador Ethernet se basaba simplemente en la MTU configurada + 12 bytes para la adición de 2 etiquetas Ethernet y el campo CRC. Los 12 bytes se agregaron a la MTU/MRU del controlador Ethernet independientemente de si había alguna etiqueta VLAN configurada en las subinterfaces.

Aquí se muestra un ejemplo con todas las versiones de Cisco IOS XR anteriores a Cisco IOS XR Release 5.1.1 y una MTU predeterminada de 1514 en una interfaz ASR 9000:

```
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>
```

En Cisco IOS XR Release 5.1.1 y versiones posteriores, la MTU y la MRU que se utilizan en el controlador de interfaz Ethernet han cambiado y ahora se basan en el número de etiquetas VLAN

que se configuran en cualquiera de las subinterfaces.

Si no se configuran etiquetas VLAN en ninguna subinterfaz, la MTU/MRU del controlador es igual a la MTU configurada en la interfaz + 4 bytes CRC, por ejemplo $1514 + 4 = 1518$ bytes.

Si se configura una VLAN en cualquier subinterfaz, la MTU/MRU del controlador es igual a la MTU configurada + 8 bytes (1 etiqueta + CRC); por ejemplo, $1514 + 8 = 1522$ bytes.

Si se configuran dos etiquetas VLAN en cualquier subinterfaz, la MTU/MRU del controlador es igual a la MTU configurada + 12 bytes (2 etiquetas + CRC); por ejemplo, $1514 + 12 = 1526$ bytes

Si QinQ con la palabra clave **any** se configura para la etiqueta second-dot1q, la MTU/MRU del controlador es igual a la MTU configurada + 8 bytes (1 etiqueta + CRC); por ejemplo, $1514 + 8 = 1522$ bytes.

Estos ejemplos muestran el comportamiento en Cisco IOS XR Release 5.1.1 y posterior en un ASR 9000:

```
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>
```

En la mayoría de las situaciones, este cambio de comportamiento en la versión 5.1.1 y posteriores no debería requerir ningún cambio en la configuración de la MTU en la interfaz.

Este cambio de comportamiento puede causar problemas en el caso de una subinterfaz configurada con una sola etiqueta VLAN, pero recibe paquetes con dos etiquetas VLAN. En esa situación, los paquetes recibidos pueden exceder la MRU en el controlador de interfaz Ethernet. Para eliminar esa condición, la MTU de la interfaz se puede incrementar en 4 bytes o la subinterfaz configurada con dos etiquetas VLAN.

La configuración automática de MTU y MRU del controlador de interfaz Ethernet en la versión 5.1.1 es la misma para los routers CRS y ASR 9000. Pero un router CRS que ejecuta la versión 5.1.1 no incluye el CRC de 4 bytes en el valor MTU y MRU que se muestra en la salida **show controller**. El comportamiento de cómo se informa no es el mismo entre CRS y ASR9000.

```
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)
```

La forma en que se visualizan la MTU y la MRU en la salida show controller en el ASR 9000 cambiará en el futuro para que los 4 bytes de CRC no se incluyan en el valor de MTU/MRU que se visualiza. Este cambio futuro se puede seguir con el ID de bug de Cisco [CSCuo93379](https://cisco.com/bug/CSCuo93379).

Convierta la configuración cuando actualice de una versión anterior a la versión 5.1.1 a la versión 5.1.1 o posterior

- MTU predeterminada:

Si había una interfaz principal sin ninguna subinterfaz y sin ningún comando `mtu` en una versión anterior a la versión 5.1.1:

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

Y esta interfaz transporta tramas dot1q o QinQ, entonces la MTU debería configurarse manualmente para "mtu 1522" en la versión 5.1.1 y posteriores:

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

Esta configuración permite transportar tramas QinQ como en las versiones anteriores. El valor de MTU se puede configurar en 1518 si sólo se transportan dot1q y no QinQ.

Si había subinterfases configuradas para dot1q o QinQ, pero con la palabra clave "any" y ninguna subinterfaz QinQ con 2 etiquetas explícitas se configuró en una versión anterior a la versión 5.1.1:

```
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
!
```

Esta configuración en la versión 5.1.1 y posteriores sólo permitirá transportar tramas con una etiqueta, por lo que la MTU también debería aumentarse manualmente en 4 bytes si se transportan tramas QinQ:

```
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
!
```

Si se configura una subinterfaz QinQ con 2 etiquetas explícitas (que no utilizan la palabra clave "any"), no hay necesidad de modificar la configuración de MTU al actualizar a la versión 5.1.1 y posteriores:

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

Si no hay una subinterfaz de transporte L2 sino solamente interfaces enrutadas L3, se espera que la configuración de MTU coincida en ambos lados y que no haya tramas más grandes que la MTU transportada. No es necesario actualizar la configuración de MTU al actualizar a la versión 5.1.1 y posteriores.

- MTU no predeterminada en la versión anterior a la versión 5.1.1:

De manera similar, cuando se configuró una MTU no predeterminada en una versión anterior a la versión 5.1.1 y no se configuró ninguna subinterfaz y las tramas dot1q o QinQ deben transportarse, el valor de MTU configurado debe aumentarse en 8 bytes al actualizar a la versión 5.1.1 o posterior.

Versión anterior a la versión 5.1.1:

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

La MTU debe aumentarse manualmente en 8 bytes cuando se actualiza a la versión 5.1.1 y posteriores:

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

El valor de MTU configurado también debe aumentarse en 4 bytes si hay una subinterfaz dot1q y ninguna subinterfaz QinQ o una subinterfaz QinQ con la palabra clave any para la segunda etiqueta dot1q.

Versión anterior a la versión 5.1.1:

```
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
!
```

Versión 5.1.1 y posteriores:

```
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
!
```

Si se configura una subinterfaz QinQ con 2 etiquetas explícitas (que no utilizan la palabra clave "any"), no hay necesidad de modificar la configuración de MTU al actualizar a la versión 5.1.1 y posteriores.

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

Si no hay una subinterfaz de transporte L2, pero sólo interfaces enrutadas L3, se espera que la configuración de MTU coincida en ambos lados y que no haya tramas más grandes que la MTU transportada. No es necesario actualizar la configuración de MTU al actualizar a la versión 5.1.1 y posteriores.

Acerca de esta traducción

Cisco ha traducido este documento combinando la traducción automática y los recursos humanos a fin de ofrecer a nuestros usuarios en todo el mundo contenido en su propio idioma.

Tenga en cuenta que incluso la mejor traducción automática podría no ser tan precisa como la proporcionada por un traductor profesional.

Cisco Systems, Inc. no asume ninguna responsabilidad por la precisión de estas traducciones y recomienda remitirse siempre al documento original escrito en inglés (insertar vínculo URL).