# Comportamento MTU sui router Cisco IOS XR e Cisco IOS

## Sommario



## Introduzione

Questo documento descrive i comportamenti dell'unità di trasmissione massima (MTU) sui router Cisco IOS® XR e confronta tali comportamenti con i router Cisco IOS. Vengono inoltre descritte le MTU sulle interfacce di layer 3 (L3) instradate e le interfacce di layer 2 VPN (L2VPN) L2 che utilizzano sia il modello EVC (Ethernet Virtual Connection) che il modello non EVC. Questo documento descrive anche importanti modifiche apportate alla configurazione automatica dell'MTU del driver dell'interfaccia Ethernet e dell'unità di ricezione massima (MRU) nella release 5.1.1 e successive.

## Premesse

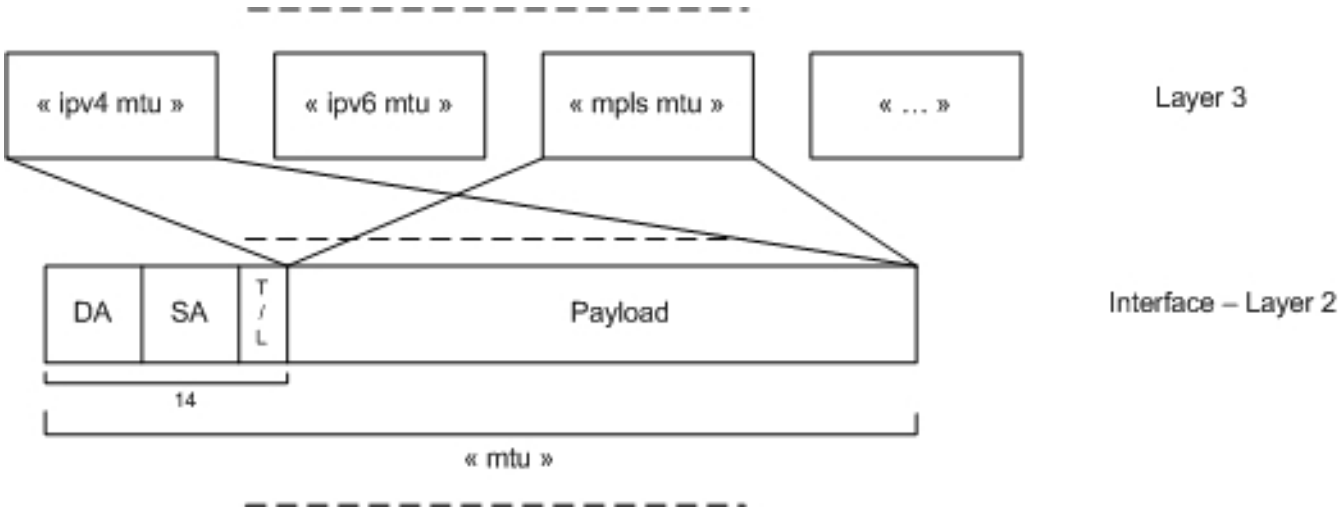
Nelle reti informatiche, l'MTU di un protocollo di comunicazione di un livello definisce le dimensioni in byte della più grande unità di dati del protocollo che il livello può trasmettere su un'interfaccia. A ogni interfaccia, livello e protocollo è associato un parametro MTU.

Le caratteristiche dell'MTU nel software Cisco IOS XR sono:

- I comandi MTU **config** e **show**, su L2 e L3, includono le dimensioni dell'intestazione del livello corrispondente. Ad esempio, il comando **mtu** che configura l'MTU L2 include 14 byte per un'interfaccia Ethernet (senza dot1q) o 4 byte per il protocollo PPP (Point-to-Point Protocol) o

il protocollo HDLC (High-Level Data Link Control). Il comando **ipv4 mtu** include 20 byte dell'intestazione IPv4.

- L'MTU di un livello superiore deve rientrare nel payload del livello inferiore. Ad esempio, se l'MTU di un'interfaccia Ethernet non dot1q è l'impostazione predefinita di 1514 byte, i protocolli di livello superiore, come Multiprotocol Label Switching (MPLS), possono avere una MTU massima di 1500 byte su quell'interfaccia. Ciò significa che è possibile inserire solo un frame MPLS da 1500 byte (incluse le etichette) all'interno del frame Ethernet. Non è possibile configurare un'MTU MPLS da 1508 byte su quell'interfaccia se si desidera consentire due tag MPLS su un pacchetto IPv4 da 1500 byte. Per trasmettere un frame MPLS da 1508 byte su un'interfaccia Ethernet, l'MTU dell'interfaccia deve essere aumentata a 1522 o un valore superiore, in modo da assicurare che il payload dell'interfaccia L2 sia sufficientemente grande da trasportare il frame MPLS.




- Nel software Cisco IOS classico (non nel software Cisco IOS XR), il comando interface **mtu** configura le dimensioni del payload L2, ma non include l'intestazione L2. Questa condizione è diversa dal software Cisco IOS XR che include il sovraccarico L2 e L3 nel comando **interface mtu**. I comandi L3 MTU, come nel caso del comando **ipv4 mtu**, configurano le dimensioni massime del pacchetto del protocollo che include l'intestazione L3. Analogamente al software Cisco IOS XR.

- L'MTU dell'interfaccia predefinita nel software Cisco IOS XR deve consentire il trasporto di un pacchetto L3 da 1500 byte. Pertanto, l'MTU predefinita è 1514 byte per un'interfaccia Ethernet principale e 1504 byte per un'interfaccia seriale.

Nella parte restante di questo documento vengono illustrate le caratteristiche dell'MTU, viene confrontato il comportamento dei software Cisco IOS e Cisco IOS XR e vengono forniti esempi per questi tipi di interfacce:

- Interfacce L3 con routing
- Sottointerfacce L3 con routing
- Interfacce L2VPN L2

# Configurazione

**Nota**: per ulteriori informazioni sui comandi menzionati in questa sezione, usare lo strumento di ricerca dei comandi (solo utenti registrati).

**Nota**: lo strumento Output Interpreter (solo utenti registrati) supporta alcuni comandi **show**. Usare lo strumento Output Interpreter per visualizzare un'analisi dell'output del comando **show.**

## Confronto tra i software Cisco IOS e Cisco IOS XR

In questa sezione vengono confrontati i comportamenti dei software Cisco IOS e Cisco IOS XR con i riferimenti alle caratteristiche MTU.

Nel software Cisco IOS, il comando **mtu** e i corrispondenti comandi **show** non includono l'intestazione L2. Usare il comando **mtu** per configurare il payload L2 sulle dimensioni massime dei pacchetti L3, compresa l'intestazione L3.

Questa procedura è diversa dal software Cisco IOS XR, in cui il comando **mtu** include l'intestazione L2 (14 byte per Ethernet o 4 byte per PPP/HDLC).

Se un router Cisco IOS è configurato con mtu *x ed è connesso a un router Cisco IOS XR, l'interfaccia corrispondente sul router Cisco IOS XR deve essere configurata con* **mtu** *x*+14 per le interfacce Ethernet o con **mtu** *x*+4 per le interfacce seriali.

I software Cisco IOS e Cisco IOS XR hanno lo stesso significato per i comandi **ipv4 mtu**, **ipv6 mtu** e **mpls mtu**; devono essere configurati con gli stessi valori.

Di conseguenza, questa è la configurazione nel software Cisco IOS su un'interfaccia Ethernet:

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

La configurazione corrispondente sul router adiacente del software Cisco IOS XR è:

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

## Interfacce L3 con routing

I valori MTU devono essere gli stessi su tutti i dispositivi connessi a una rete L2. In caso contrario, potrebbero essere segnalati i seguenti sintomi:

- Non vengono visualizzate le adiacenze IS-IS (Intermediate System-to-Intermediate System). Per impostazione predefinita, l'IS-IS utilizza la spaziatura interna; pertanto, gli hellos potrebbero essere caratterizzati come giants e potrebbero essere scartati quando un router ha un valore MTU inferiore ai valori degli altri router.
- Le adiacenze OSPF (Open Shortest Path First) sono bloccate nello stato Exstart o Exchange, perché i pacchetti DBD (Large Database Descriptor) potrebbero essere caratterizzati come

giganti e quindi essere eliminati. Quando i pacchetti vengono ricevuti su un router con un valore MTU inferiore, i database non vengono sincronizzati.
- Il traffico di dati è caratterizzato da giganti e viene scartato quando viene ricevuto su un dispositivo con un valore MTU inferiore a quello del dispositivo trasmittente.
- Il throughput è basso quando i pacchetti di grandi dimensioni vengono scartati. In caso di rilevamento dell'MTU del percorso, la sessione TCP può recuperare quando i pacchetti di grandi dimensioni vengono scartati, ma questo influisce sulla velocità effettiva.

## MTU predefinita

Questa sezione analizza l'MTU predefinita di un'interfaccia indirizzata quando il comando **mtu** non è configurato:

```
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_GETHERNET
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#
```

Nell'esempio, l'MTU predefinita dell'interfaccia L2 è 1514 byte e include 14 byte di intestazione Ethernet. I 14 byte vengono conteggiati da 6 byte di indirizzo MAC di destinazione, 6 byte di indirizzo MAC di origine e 2 byte di tipo o lunghezza. Sono esclusi il preambolo, il delimitatore di fotogramma, 4 byte di sequenza di controllo del fotogramma (FCS) e l'intervallo tra fotogrammi. Per un frame PPP o HDLC, vengono presi in considerazione i 4 byte dell'intestazione L2; quindi l'MTU predefinita dell'interfaccia è 1504 byte.

I protocolli figlio L3 ereditano la MTU dal payload dell'MTU padre. Se si sottraggono 14 byte di un'intestazione L2 da una MTU L2 di 1514 byte, il payload L2 è di 1500 byte. Questa diventa l'MTU dei protocolli L3. IPv4, IPv6, MPLS e il servizio di rete senza connessione (CLNS) ereditano la MTU di 1500 byte. Di conseguenza, per impostazione predefinita, un'interfaccia Cisco IOS XR Ethernet può trasportare un pacchetto L3 da 1500 byte, ossia lo stesso valore di defaultIt su un'interfaccia Cisco IOS Ethernet.

## MTU non predefinita

In questa sezione viene mostrato come configurare una **mtu mpls** di 1508 per inviare un pacchetto IPv4 di 1500 byte con due tag MPLS di 4 byte ciascuno, sopra il pacchetto:

```
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_GETHERNET
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#
```

Anche se il comando **mpls mtu 1508** è stato eseguito, non viene applicato, in quanto MPLS ha ancora una MTU di 1500 byte nel comando **show**. Infatti, i protocolli figlio L3 non possono avere una MTU più grande del payload dell'interfaccia L2 padre.

Per consentire due etichette su un pacchetto IP da 1500 byte, è necessario:

- Configurare una MTU dell'interfaccia L2 di 1522 byte, in modo che tutti i protocolli figlio (incluso MPLS) ereditino una MTU di 1508 byte (1522 - 14 = 1508).
- Ridurre la MTU dei protocolli L3 a 1500 byte, in modo che solo MPLS possa superare i 1500 byte.

```
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_GETHERNET
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#
```

Questa configurazione consente di inviare pacchetti IPv4 e IPv6 da 1500 byte e pacchetti MPLS da 1508 byte (un pacchetto da 1500 byte con due tag nella parte superiore).

## Sottointerfacce L3 con routing

Queste caratteristiche si applicano alle sottointerfacce L3 con routing.

L'MTU di una sottointerfaccia con routing eredita l'MTU dell'interfaccia principale padre; aggiungere 4 byte per ciascun tag VLAN configurato sull'interfaccia secondaria. Pertanto, sono disponibili 4 byte per un'interfaccia secondaria dot1q e 8 byte per un'interfaccia secondaria tunneling IEEE 802.1Q (QinQ).

Di conseguenza, i pacchetti L3 delle stesse dimensioni possono essere inoltrati sia sull'interfaccia principale che sull'interfaccia secondaria.

il comando **mtu** può essere configurato nell'interfaccia secondaria, ma viene applicato solo se è inferiore o uguale all'MTU ereditata dall'interfaccia principale.

Questo è un esempio di MTU dell'interfaccia principale di 2000 byte:

```
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#
```

Nei comandi **show**, l'MTU dell'interfaccia secondaria è 2004; aggiungere 4 byte alla MTU dell'interfaccia principale perché è presente un tag dot1q configurato nell'interfaccia secondaria.

Tuttavia, la MTU dei pacchetti IPv4 e IPv6 è ancora la stessa dell'interfaccia principale (1986). Infatti, l'MTU dei protocolli L3 è ora calcolata come: 2004 - 14 - 4 = 1986.

Il comando **mtu** può essere configurato nella sottointerfaccia, ma l'MTU configurata viene applicata solo se è inferiore o uguale alla MTU ereditata dall'interfaccia principale (4 byte più grande della MTU dell'interfaccia principale).

Quando l'MTU dell'interfaccia secondaria è più grande dell'MTU ereditata, non viene applicata, come mostrato di seguito:

```
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#
```

Pertanto, è possibile usare solo il comando **mtu** per ridurre il valore MTU ereditato dall'interfaccia principale.

Analogamente, è possibile usare anche i comandi MTU dei protocolli L3 (IPv4, IPv6, MPLS) per ridurre il valore della MTU L3 ereditata dal payload L2 della sottointerfaccia. L'MTU del protocollo L3 non ha effetto quando è configurata su un valore che non rientra nel payload dell'MTU L2.

## Interfaccia L2VPN L2

L'MTU di una L2VPN è importante perché il protocollo LDP (Label Distribution Protocol) non solleva uno pseudowire (PW) quando le MTU dei circuiti di collegamento su ciascun lato di una PW non sono le stesse.

Di seguito è riportato un comando **show** per illustrare che un PW L2VPN rimane inattivo quando le MTU non corrispondono:

```
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 0xfffe0001
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#
```

Nell'esempio, notare che i peer del provider MPLS L2VPN (PE) su ciascun lato devono segnalare lo stesso valore MTU per riattivare il PW.

L'MTU segnalata da MPLS LDP non include il sovraccarico L2. Questa condizione è diversa dai comandi XR interface config e show che includono il sovraccarico L2. La MTU sull'interfaccia secondaria è di 2018 byte (ereditata dall'interfaccia principale di 2014 byte), ma LDP ha segnalato una MTU di 2000 byte.  Di conseguenza, sottrae 18 byte (14 byte di intestazione Ethernet + 4 byte di tag 1 dot1q) dall'intestazione L2.

Èimportante capire come ciascun dispositivo calcola i valori MTU dei circuiti di collegamento per risolvere i problemi di MTU. Dipende da parametri quali fornitore, piattaforma, versione del software e configurazione.

## EVC (ASR9000)

Cisco ASR serie 9000 Aggregation Services Router utilizza il modello di infrastruttura EVC, che consente una corrispondenza VLAN flessibile su interfacce e sottointerfacce L2VPN.

Le interfacce EVC L2VPN L2 hanno le seguenti caratteristiche:

- e consentono di configurare uno o più tag con il comando **encapsulation**.
- Per impostazione predefinita, e solo con il comando **encapsulation**, i tag vengono conservati e trasportati sui PW. Di conseguenza, non è necessario rimuovere i tag per impostazione predefinita, come è necessario fare sulle piattaforme non EVC.
- Usate il comando **rewrite** quando decidete di inserire i tag in arrivo o di inserire altri tag sopra il frame in arrivo.

Per calcolare l'MTU dell'interfaccia secondaria, prendere l'MTU dell'interfaccia principale (quella predefinita o quella configurata manualmente sull'interfaccia principale) e aggiungere 4 byte per ciascun tag VLAN configurato con il comando **encapsulation**. Vedere Comandi di incapsulamento EFP specifici.

Quando è presente un comando **mtu** sotto l'interfaccia secondaria, ha effetto solo se è inferiore all'MTU calcolata. Il comando **rewrite** non influenza l'MTU dell'interfaccia secondaria.

Di seguito è riportato un esempio:

```
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)
```

In questo esempio, l'MTU dell'interfaccia principale è di 2014 byte. Aggiungere 8 byte perché ci sono due tag configurati sotto l'interfaccia secondaria.

Se si configura l'**MTU di 2026** byte nell'interfaccia secondaria, questa non viene applicata perché è più grande della MTU dell'interfaccia secondaria ereditata dall'interfaccia principale (2022). Di conseguenza, è possibile configurare solo una MTU dell'interfaccia secondaria inferiore a 2022 byte.

In base a questa MTU dell'interfaccia secondaria, calcolare l'MTU del payload LDP MPLS segnalato al router adiacente e verificare che sia identica a quella calcolata dal server PE L2VPN remoto. In questo caso, entra in gioco il comando **rewrite**.

Per calcolare l'MTU del payload MPLS LDP, prendere la MTU della sottointerfaccia, quindi:

1. Sottrarre 14 byte dall'intestazione Ethernet.
2. Sottrarre 4 byte per ciascun tag visualizzato nel comando **rewrite** configurato nell'interfaccia secondaria.
3. Aggiungere 4 byte per ogni tag inserito nel comando **rewrite** configurato nell'interfaccia secondaria.

Questo è lo stesso esempio della configurazione QinQ sul 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)
```

Questo è il calcolo della MTU del payload LDP MPLS:

1. Valore MTU dell'MTU della sottointerfaccia: 2022 byte
2. Sottrarre 14 byte dell'intestazione Ethernet: 2022 - 14 = 2008 byte
3. Sottrarre 4 byte per ogni tag aperto nel **riscrivere** comando: 2008 - 4 * 2 = 2000

Verificare che il lato remoto annunci un payload MPLS LDP di 2000 byte. In caso contrario, regolare le dimensioni dell'MTU del circuito di collegamento locale o remoto (CA) in modo che corrispondano.

```
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
```

#### Comandi specifici di incapsulamento Ethernet Flow Point (EFP)

Poiché questi incapsulamenti contano come tag zero corrispondenti, non aumentano l'MTU dell'interfaccia secondaria:

- **incapsulamento senza tag**
- **valore predefinito di incapsulamento**

I seguenti modificatori di incapsulamento non influiscono sul numero di tag richiesti per calcolare l'MTU della sottointerfaccia:

- nativo
- payload-ethertype

- esatto
- cos
- source-mac in entrata o destination-mac in entrata

**l'incapsulamento [dot1q|dot1ad] con tag di priorità** viene conteggiato come un singolo tag.

La parola chiave 'any' usata come corrispondenza del tag più interna non aumenta l'MTU dell'interfaccia secondaria.

- **l'incapsulamento dot1q any** non aumenta l'MTU della sottointerfaccia.
- **l'incapsulamento dot1e 10 dot1q any** viene considerato come un unico tag; aumenta l'MTU della sottointerfaccia di 4 byte.
- **l'incapsulamento dot1e qualsiasi dot1q 7** viene considerato come due tag; aumenta l'MTU della sottointerfaccia di 8 byte.

Gli intervalli di ID VLAN incrementano l'MTU della sottointerfaccia:

- **l'incapsulamento dot1q 10-100** viene conteggiato come un unico tag; aumenta l'MTU della sottointerfaccia di 4 byte.

Il sovraccarico MTU dell'incapsulamento di una EFP con corrispondenza disgiuntiva viene considerato come la MTU dell'elemento più alto.

- **incapsulamento dot1q 10-100, senza tag** viene considerato come un tag perché l'intervallo 10-100 è l'elemento più alto.

### Non EVC (XR 12000 e CRS)

Router come Cisco XR serie 12000 e Carrier Routing System (CRS) utilizzano la configurazione tradizionale per la corrispondenza VLAN sulle sottointerfacce. Queste caratteristiche si applicano alle interfacce L2VPN L2 su CRS e su router XR 12000 non conformi al modello EVC:

- Sulle piattaforme non EVC, i tag dot1q o dot1ad in entrata vengono automaticamente rimossi quando vengono ricevuti su una sottointerfaccia di trasporto L2.
- Quando si calcolano le dimensioni del payload per il segnale LDP MPLS, sottrarre le dimensioni dei tag dalla MTU della sottointerfaccia, come mostrato nel comando **show interface**.
- Analogamente a quanto avviene per le sottointerfacce instradate.
- La sottointerfaccia eredita la MTU dall'interfaccia principale; aggiungere i 4 byte di ciascun tag alla MTU dell'interfaccia principale per calcolare la MTU della sottointerfaccia. Ad esempio, se una sottointerfaccia QinQ ha 2 tag dot1q, per impostazione predefinita la MTU dell'interfaccia secondaria è più grande di 8 byte della MTU dell'interfaccia principale.
- Èpossibile usare il comando **mtu** anche nella sottointerfaccia, ma viene usato solo per ridurre la MTU della sottointerfaccia, che viene ereditata dalla MTU dell'interfaccia principale.

Ecco alcuni esempi che illustrano queste caratteristiche.

Nell'esempio viene mostrato come configurare un'interfaccia secondaria non 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#
```

Le piattaforme non EVC usano i comandi **dot1q vlan** o **dot1ad vlan** al posto dei comandi **encapsulation** e **rewrite** delle piattaforme EVC (ASR9000).

Se non si configura un MTU in modo esplicito sull'interfaccia principale o secondaria, è possibile ricevere un pacchetto L3 da 1500 byte per impostazione predefinita:

```
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#
```

L'MTU della sottointerfaccia viene calcolata dall'MTU dell'interfaccia principale (1514); aggiungere 4 byte per ciascun tag dot1q. Poiché sull'interfaccia secondaria è configurato un tag con il comando **dot1q vlan 201**, aggiungere 4 byte a 1514 per una MTU di 1518 byte.

La MTU del payload corrispondente in MPLS LDP è di 1500 byte, poiché i 14 byte dell'intestazione Ethernet non vengono conteggiati e il tag one dot1q viene scaricato automaticamente dalla piattaforma non EVC quando supera il 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
```

Se si aumenta la MTU dell'interfaccia principale a 2014 byte, la MTU dell'interfaccia secondaria viene aumentata di conseguenza:

```
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
```

Quindi, per calcolare l'MTU LDP MPLS, sottrarre 14 byte dall'intestazione Ethernet e aggiungere 4 byte per ciascun tag configurato nell'interfaccia secondaria.

# Configurazione automatica MTU e MRU del driver dell'interfaccia Ethernet

Sulle interfacce Ethernet, il driver di interfaccia è configurato con una MTU e una MRU basata sulla configurazione dell'MTU dell'interfaccia.

L'MTU e l'MRU configurati sul driver dell'interfaccia Ethernet possono essere visualizzati con il comando **show controller <interface>all**.

Nelle versioni precedenti a Cisco IOS XR release 5.1.1, l'MTU e l'MRU sul driver di interfaccia Ethernet sono stati configurati automaticamente in base alla configurazione dell'MTU di Cisco IOS XR sull'interfaccia.

L'MTU/MRU configurata sul driver Ethernet si basava semplicemente sulla MTU configurata + 12 byte per l'aggiunta di 2 tag Ethernet e del campo CRC. I 12 byte sono stati aggiunti all'MTU/MRU del driver Ethernet indipendentemente dalla presenza di tag VLAN configurati sulle sottointerfacce.

Di seguito è riportato un esempio di tutte le versioni di Cisco IOS XR precedenti alla versione 5.1.1 e di una MTU predefinita di 1514 su un'interfaccia 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>
```

In Cisco IOS XR versione 5.1.1 e successive, l'MTU e l'MRU utilizzati sul driver dell'interfaccia Ethernet sono stati modificati e ora si basano sul numero di tag VLAN configurati su una delle sottointerfacce.

Se non sono configurati tag VLAN sulle sottointerfacce, l'MTU/MRU del driver è uguale all'MTU configurata sull'interfaccia + 4 byte CRC, ad esempio 1514 + 4 = 1518 byte.

Se su una delle sottointerfacce è configurata una VLAN, l'MTU/MRU del driver è uguale alla MTU configurata + 8 byte (1 tag + CRC), ad esempio 1514 + 8 = 1522 byte.

Se due tag VLAN sono configurati su una qualsiasi sottointerfaccia, l'MTU/MRU del driver è uguale alla MTU configurata + 12 byte (2 tag + CRC), ad esempio 1514 + 12 = 1526 byte

Se QinQ con la parola chiave **any** è configurato per il tag second-do1q, l'MTU/MRU del driver è uguale all'MTU configurata + 8 byte (1 tag + CRC), ad esempio 1514 + 8 = 1522 byte.

Gli esempi riportati di seguito mostrano il comportamento di Cisco IOS XR release 5.1.1 e successive su 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>
```

Nella maggior parte delle situazioni, il cambiamento di questo comportamento nella release 5.1.1 e successive non deve richiedere alcuna modifica alla configurazione dell'MTU sull'interfaccia.

Questo cambiamento di comportamento può causare problemi in una sottointerfaccia configurata con un singolo tag VLAN, ma riceve pacchetti con due tag VLAN. In questa situazione, i pacchetti ricevuti possono superare l'MRU sul driver dell'interfaccia Ethernet. Per eliminare questa condizione, l'MTU dell'interfaccia può essere aumentata di 4 byte o la sottointerfaccia può essere configurata con due tag VLAN.

La configurazione automatica dell'MTU e dell'MRU del driver dell'interfaccia Ethernet nella release 5.1.1 è la stessa per un router CRS e ASR 9000. Tuttavia, un router CRS con versione 5.1.1 non include il CRC da 4 byte nel valore MTU e MRU visualizzato nell'output **show controller**.  Il comportamento della modalità di segnalazione non è lo stesso tra CRS e 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)
```

Il modo in cui l'MTU e l'MRU vengono visualizzati nell'output show controller sull'ASR 9000 verrà modificato in futuro in modo che i 4 byte di CRC non vengano inclusi nel valore MTU/MRU visualizzato. Questa modifica futura può essere rilevata con l'ID bug Cisco CSCuo93379.

## Convertire la configurazione quando si esegue l'aggiornamento da una release precedente alla 5.1.1 alla release 5.1.1 o successive

- MTU predefinita:

se l'interfaccia principale era senza alcuna sottointerfaccia e senza comando **mtu** in una versione precedente alla 5.1.1:

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

Inoltre, questa interfaccia trasporta frame dot1q o QinQ, quindi l'MTU deve essere configurata manualmente su "mtu 1522" nella versione 5.1.1 e successive:

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

Questa configurazione consente il trasporto dei frame QinQ, come nelle versioni precedenti. Il valore MTU può essere configurato su 1518 se si devono trasportare solo dot1q e non QinQ.

Se erano presenti sottointerfacce configurate per dot1q o QinQ, ma con la parola chiave "any" (qualsiasi) e nessuna sottointerfaccia QinQ con 2 tag espliciti è stata configurata in una release precedente alla 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
!
```

Questa configurazione nella release 5.1.1 e successive consente solo di trasportare i frame con un tag, quindi se i frame QinQ devono essere trasportati, l'MTU deve essere aumentata manualmente di 4 byte:

```
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
!
```

Se è stata configurata un'interfaccia secondaria QinQ con 2 tag espliciti (che non usano la parola chiave "any"), non è necessario modificare la configurazione MTU quando si esegue l'aggiornamento alla release 5.1.1 e successive:

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

Se non esiste una sottointerfaccia di trasporto L2 ma solo interfacce con routing L3, si prevede che la configurazione MTU corrisponda su entrambi i lati e che non vi siano frame più grandi della MTU trasportata. non è necessario aggiornare la configurazione MTU quando si esegue l'aggiornamento alla release 5.1.1 e successive.

- MTU non predefinita nella release precedente alla 5.1.1:

Analogamente, se in una release precedente alla 5.1.1 è stata configurata una MTU non predefinita e non è stata configurata alcuna sottointerfaccia ed è necessario trasportare i frame dot1q o QinQ, il valore MTU configurato deve essere aumentato di 8 byte quando si esegue l'aggiornamento alla release 5.1.1 o successive.

Release precedenti alla release 5.1.1:

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

Quando si esegue l'aggiornamento alla release 5.1.1 e successive, l'MTU deve essere aumentata manualmente di 8 byte:

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

Il valore MTU configurato deve essere aumentato di 4 byte se c'è una sottointerfaccia dot1q e nessuna sottointerfaccia QinQ o una sottointerfaccia QinQ con la parola chiave any per il secondo tag dot1q.

Release precedenti alla release 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
!
```

Versione 5.1.1 e successive:

```
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
!
```

Se è stata configurata un'interfaccia secondaria QinQ con 2 tag espliciti (che non usano la parola chiave "any"), non è necessario modificare la configurazione MTU quando si esegue l'aggiornamento alla release 5.1.1 e successive.

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

Se non esiste una sottointerfaccia di trasporto L2, ma solo interfacce con routing L3, si prevede che la configurazione MTU corrisponda su entrambi i lati e che non vi siano frame più grandi della MTU trasportata. non è necessario aggiornare la configurazione MTU quando si esegue l'aggiornamento alla release 5.1.1 e successive.

Informazioni su questa traduzione

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