

# Distribuzione del peso CEF nella condivisione del carico a costi diversi

## Sommario

[Introduzione](#)

[Prerequisiti](#)

[Requisiti](#)

[Componenti usati](#)

[Panoramica di UCMP](#)

[Configurazioni iniziali](#)

[Pesi/carico metrica](#)

[Determinazione varianza UCMP](#)

[Comprensione dei pesi](#)

[Determinazione dei valori di peso](#)

[Peso](#)

[Peso normalizzato](#)

[Manipolazione dei rapporti peso/carico CEF](#)

[Esempio 1: Rapporto peso/carico 26/5](#)

[Esempio 2: Rapporto peso/carico di 30/1](#)

## Introduzione

Questo documento descrive gli aspetti di Comprensione, configurazione e verifica dei percorsi multipli diseguali in IOS-XR. Inoltre, vengono presentati esempi di manipolazioni del peso per mostrare in che modo la metrica del percorso verso una destinazione influisce sul carico di un collegamento.

## Prerequisiti

Non sono previsti prerequisiti per questo documento.

## Requisiti

Gli esempi seguenti sono basati su IOS-XR 6.4.1.

## Componenti usati

Le informazioni discusse in questo documento fanno riferimento a dispositivi usati in uno specifico ambiente di emulazione. Su tutti i dispositivi menzionati nel documento la configurazione è stata ripristinata ai valori predefiniti. Se la rete è operativa, valutare attentamente eventuali conseguenze derivanti dall'uso dei comandi.

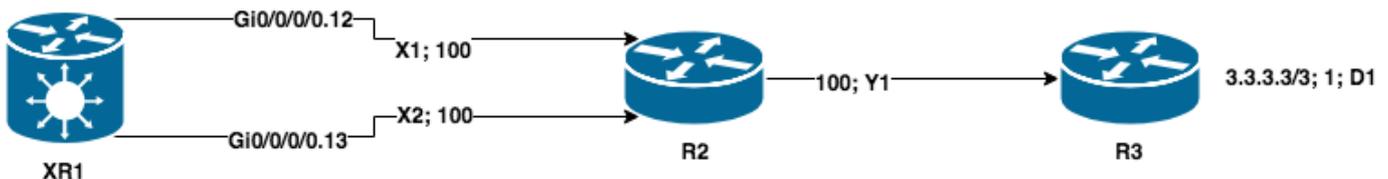
# Panoramica di UCMP

Il bilanciamento del carico UCMP (Unequal Cost Multipath) consente di bilanciare il carico in modo proporzionale su più percorsi, con costi diversi. In genere, per i percorsi con larghezza di banda più elevata è configurata una metrica IGP (Interior Gateway Protocol) inferiore, in modo che formino i percorsi IGP più brevi.

Con il bilanciamento del carico UCMP abilitato, i protocolli possono utilizzare percorsi di larghezza di banda ancora più bassi o percorsi di costo maggiore per il traffico e possono installare questi percorsi nel FIB (Forwarding Information Base). Questi protocolli installano ancora più percorsi alla stessa destinazione in FIB, ma a ogni percorso sarà associato un 'load metric/weight'. FIB utilizza questa metrica/peso del carico per decidere la quantità di traffico da inviare su un percorso con larghezza di banda superiore e la quantità di traffico da inviare su un percorso con larghezza di banda inferiore.

Tradizionalmente, EIGRP è stato l'unico IGP che supporta la funzione UCMP, ma in IOS-XR UCMP è supportato per tutti gli IGP, il routing statico e BGP. In questo documento verrà illustrata la funzionalità UCMP utilizzando OSPF come base per gli esempi, ma le informazioni qui riportate si applicano anche all'IS-IS e ad altri protocolli compatibili con UCMP.

## Diagramma topologico



## Configurazioni iniziali

```
XR1
!
hostname XR1
!
interface GigabitEthernet0/0/0/0.12
description TO R2
ipv4 address 12.0.0.1 255.255.255.0
encapsulation dot1q 12
!
interface GigabitEthernet0/0/0/0.13
description TO R2
ipv4 address 13.0.0.1 255.255.255.0
encapsulation dot1q 13
! router ospf 1 address-family ipv4 area 0 ! interface GigabitEthernet0/0/0/0.12      cost 100
!
! interface GigabitEthernet0/0/0/0.13
  cost 100
!
!
!
end

R2
!
hostname R2
```

```

!
interface Ethernet0/0.12
  description TO XR1
  encapsulation dot1Q 12
  ip address 12.0.0.2 255.255.255.0
!
interface Ethernet0/0.13
  description TO XR1
  encapsulation dot1Q 13
  ip address 13.0.0.2 255.255.255.0
!
interface Ethernet0/1
  description TO R3
  ip address 172.16.23.2 255.255.255.0
  ip ospf cost 100
!
!
router ospf 1
  network 0.0.0.0 255.255.255.255 area 0
!
end

```

### R3

```

!
hostname R3
!
interface Loopback0
  description FINAL_DESTINATION
  ip address 3.3.3.3 255.255.255.255
!
interface Ethernet0/0
  description TO R2
  ip address 172.16.23.3 255.255.255.0
!
router ospf 1
  network 0.0.0.0 255.255.255.255 area 0
!
end

```

## Pesi/carico metrica

In IOS-XR, quando si installano più percorsi verso una destinazione, alla destinazione viene assegnato un valore di peso che indica la distribuzione del carico per un particolare collegamento. Questo valore è inversamente proporzionale alla metrica del percorso alla destinazione, più alto è il costo, più basso è il peso assegnato. Questo consente al CEF di eseguire in modo intelligente la condivisione del carico dei collegamenti durante il routing alle destinazioni.

Quando si installano i percorsi ECMP, i valori di peso assegnati sono sempre impostati su 0 per tutti i percorsi, il che significa che il traffico viene condiviso equamente. Se si controlla CEF, è possibile verificare che siano stati assegnati pesi pari a 0 per ciascun percorso.

```
RP/0/RP0/CPU0:XR1#show cef 3.3.3.3/32 detail
```

```

3.3.3.3/32, version 87, internal 0x1000001 0x0 (ptr 0xd689b50) [1], 0x0 (0xd820648), 0x0 (0x0)
Updated Nov 11 22:15:58.953
remote adjacency to GigabitEthernet0/0/0/0.12
Prefix Len 32, traffic index 0, precedence n/a, priority 1
gateway array (0xd6b32f8) reference count 2, flags 0x0, source rib (7), 0 backups
      [3 type 3 flags 0x8401 (0xd759758) ext 0x0 (0x0)]
LW-LDI[type=3, refc=1, ptr=0xd820648, sh-ldi=0xd759758]

```

```
gateway array update type-time 1 Nov 11 22:15:58.953
```

```
LDI Update time Nov 11 22:15:58.953
```

```
LW-LDI-TS Nov 11 22:15:58.953
```

```
via 12.0.0.2/32, GigabitEthernet0/0/0/0.12, 4 dependencies, weight 0, class 0 [flags 0x0]
```

```
path-idx 0 NHID 0x0 [0xe14b0a0 0x0]
```

```
next hop 12.0.0.2/32
```

```
remote adjacency
```

```
via 13.0.0.2/32, GigabitEthernet0/0/0/0.13, 4 dependencies, weight 0, class 0 [flags 0x0]
```

```
path-idx 1 NHID 0x0 [0xe14b128 0x0]
```

```
next hop 13.0.0.2/32
```

```
remote adjacency
```

```
Load distribution: 0 1 (refcount 3)
```

Hash	OK	Interface	Address
0	Y	GigabitEthernet0/0/0/0.12	remote
1	Y	GigabitEthernet0/0/0/0.13	remote

## Determinazione varianza UCMP

Se si desidera abilitare UCMP, è innanzitutto necessario impostare il costo in modo diverso su XR1. Per questo motivo, il costo verrà impostato come indicato di seguito:

```
router ospf 1
address-family ipv4
area 0
interface Loopback0
!
interface GigabitEthernet0/0/0/0.12
cost 50
!
interface GigabitEthernet0/0/0/0.13
cost 100
!
!
end
```

```
RP/0/RP0/CPU0:XR1#show route 3.3.3.3/32
```

```
Routing entry for 3.3.3.3/32
Known via "ospf 1", distance 110, metric 151, type intra area
Installed Nov 11 22:32:48.289 for 00:00:32
Routing Descriptor Blocks
12.0.0.2, from 3.3.3.3, via GigabitEthernet0/0/0/0.12
Route metric is 151
No advertising protos.
```

Per considerare altri percorsi per UCMP, è necessario determinare se sono idonei. IOS-XR utilizza un criterio percentuale per IS-IS e OSPF, basato sul comando `ucmp variance <value>` router process. Le due strade che abbiamo sono:

metrica percorso 1 (pm1) = 151

path metric 2 (pm2) = 201

Gli hop successivi senza loop verranno installati in base a  $UCMP \leq (Varianza * Metrica\ percorso\ primario) / 100$ .

Il valore di incremento del percorso primario per raggiungere la metrica del percorso peggiore

(pm2) in questo caso è 134% di 151, ovvero 202. Questo è il valore di varianza esatto da configurare per rendere il percorso idoneo.

```
!  
router ospf 1  
  ucmp variance 134  
!  
  
RP/0/RP0/CPU0:XR1#show route 3.3.3.3/32  
  
Routing entry for 3.3.3.3/32  
  Known via "ospf 1", distance 110, metric 151, type intra area  
  Installed Nov 11 22:36:45.720 for 00:00:09  
  Routing Descriptor Blocks  
    12.0.0.2, from 3.3.3.3, via GigabitEthernet0/0/0/0.12  
      Route metric is 151, Wt is 4294967295  
    13.0.0.2, from 3.3.3.3, via GigabitEthernet0/0/0/0.13  
      Route metric is 151, Wt is 3226567396  
  No advertising protos.
```

### [Spoiler](#)

**Nota:** Il valore della varianza non ha alcun impatto sui risultati del peso. In questo caso una varianza minima di 134 o una varianza di 10000 (valore massimo) avrebbe portato agli stessi risultati di ponderazione, invece, i valori di costo sono quelli che influenzano i pesi risultanti, in quanto sono inversamente proporzionali l'uno all'altro.

**Nota:** Il valore della varianza non ha alcun impatto sui risultati del peso. In questo caso una varianza minima di 134 o una varianza di 10000 (valore massimo) avrebbe portato agli stessi risultati di ponderazione, invece, i valori di costo sono quelli che influenzano i pesi risultanti, in quanto sono inversamente proporzionali l'uno all'altro.

## Comprensione dei pesi

In IOS-XR esistono due tipi diversi di pesi: **peso** e **pesi normalizzati**. L'uso di questi bucket si basa sul numero di bucket di hash supportati su una particolare piattaforma, XRv9000 supporta 32 bucket di hash, ASR 9000 e CRS-X supportano rispettivamente 64 bucket di hash. Ciò significa che, quando il router programma i valori del peso, il peso non può superare il limite dell'intervallo di hash della piattaforma in questione. Per verificare i pesi normalizzati programmati, usare il comando **show cef <prefix> detail location<location>**. In base ai valori di costo impostati, abbiamo una distribuzione del carico di 18, 13, il che significa che sono stati assegnati 31 bucket di hash (18+13).

```
RP/0/RP0/CPU0:XR1#show cef 3.3.3.3/32 detail  
  
3.3.3.3/32, version 23, internal 0x1000001 0x0 (ptr 0xd3ecb50) [1], 0x0 (0xd583610), 0x0 (0x0)  
  Updated Nov 11 22:36:45.723  
  remote adjacency to GigabitEthernet0/0/0/0.12  
  Prefix Len 32, traffic index 0, precedence n/a, priority 1  
  gateway array (0xd4163d8) reference count 1, flags 0x0, source rib (7), 0 backups  
    [2 type 3 flags 0x8401 (0xd4bc7b8) ext 0x0 (0x0)]  
  LW-LDI[type=3, refc=1, ptr=0xd583610, sh-ldi=0xd4bc7b8]  
  gateway array update type-time 1 Nov 11 22:36:45.723  
  LDI Update time Nov 11 22:36:45.729  
  LW-LDI-TS Nov 11 22:36:45.729
```

via 12.0.0.2/32, GigabitEthernet0/0/0/0.12, 6 dependencies, weight 4294967295, class 0 [flags 0x0]

path-idx 0 NHID 0x0 [0xe14b1b0 0x0]  
next hop 12.0.0.2/32  
remote adjacency

via 13.0.0.2/32, GigabitEthernet0/0/0/0.13, 6 dependencies, weight 3226567396, class 0 [flags 0x0]

path-idx 1 NHID 0x0 [0xe14b128 0x0]  
next hop 13.0.0.2/32  
remote adjacency

**Weight distribution:**

slot 0, weight 4294967295, normalized\_weight 18, class 0  
slot 1, weight 3226567396, normalized\_weight 13, class 0

Load distribution: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 (refcount

2)

Hash	OK	Interface	Address
0	Y	GigabitEthernet0/0/0/0.12	remote
1	Y	GigabitEthernet0/0/0/0.12	remote
2	Y	GigabitEthernet0/0/0/0.12	remote
3	Y	GigabitEthernet0/0/0/0.12	remote
4	Y	GigabitEthernet0/0/0/0.12	remote
5	Y	GigabitEthernet0/0/0/0.12	remote
6	Y	GigabitEthernet0/0/0/0.12	remote
7	Y	GigabitEthernet0/0/0/0.12	remote
8	Y	GigabitEthernet0/0/0/0.12	remote
9	Y	GigabitEthernet0/0/0/0.12	remote
10	Y	GigabitEthernet0/0/0/0.12	remote
11	Y	GigabitEthernet0/0/0/0.12	remote
12	Y	GigabitEthernet0/0/0/0.12	remote
13	Y	GigabitEthernet0/0/0/0.12	remote
14	Y	GigabitEthernet0/0/0/0.12	remote
15	Y	GigabitEthernet0/0/0/0.12	remote
16	Y	GigabitEthernet0/0/0/0.12	remote
17	Y	GigabitEthernet0/0/0/0.12	remote
18	Y	GigabitEthernet0/0/0/0.13	remote
19	Y	GigabitEthernet0/0/0/0.13	remote
20	Y	GigabitEthernet0/0/0/0.13	remote
21	Y	GigabitEthernet0/0/0/0.13	remote
22	Y	GigabitEthernet0/0/0/0.13	remote
23	Y	GigabitEthernet0/0/0/0.13	remote
24	Y	GigabitEthernet0/0/0/0.13	remote
25	Y	GigabitEthernet0/0/0/0.13	remote
26	Y	GigabitEthernet0/0/0/0.13	remote
27	Y	GigabitEthernet0/0/0/0.13	remote
28	Y	GigabitEthernet0/0/0/0.13	remote
29	Y	GigabitEthernet0/0/0/0.13	remote
30	Y	GigabitEthernet0/0/0/0.13	remote

Come si può osservare, la somma del peso normalizzato rappresenta la quantità di bucket di hash assegnati dalla piattaforma, in questo caso, non possiamo mai superare 32 bucket di hash, secondo il limite di questa particolare piattaforma. Il peso del percorso primario (pm1) è sempre impostato su 4294967295, ovvero il peso massimo ( $2^{32} - 1$ ).

## Determinazione dei valori di peso

### Peso

È possibile calcolare facilmente i pesi con la formula  $\text{peso} = \text{miglior costo} / \text{peggiore costo} *$

**4294967295.** Ad esempio, i pesi per il percorso 1 e il percorso 2 vengono calcolati di seguito:

Weight\_path\_1 = sempre impostato su 4294967295

Weight\_path\_2 =  $151 / 201 * 4294967295 = 3226567470$

### [Spoiler](#)

**Nota: Durante il calcolo dei valori può verificarsi una perdita di precisione, come nel caso dei calcoli a virgola mobile. È necessario installare numeri interi in RIB e FIB.**

Nota: Durante il calcolo dei valori può verificarsi una perdita di precisione, come nel caso dei calcoli a virgola mobile. È necessario installare numeri interi in RIB e FIB.

## **Peso normalizzato**

Come accennato, non possiamo installare nella tabella CEF valori di peso che superano la quantità di bucket di hash da una piattaforma, a causa di questo abbiamo bisogno di normalizzare i pesi prima di programmarli in hardware. La piattaforma calcola i pesi normalizzati in base alla formula **Peso normalizzato = (Peso percorso/Peso totale) \* Dimensione massima del periodo fisso**. In base all'esempio, è possibile calcolare quanto segue:

$normalized\_weight\_1 = (4294967295 * 32) / (3226567396 + 4294967295) = 18$

$normalized\_weight\_2 = (3226567396 * 32) / (3226567396 + 4294967295) = 13$

### [Spoiler](#)

**Nota: Quando G.C.D è uguale a 1, viene utilizzato il metodo precedente, altrimenti se G.C.D ≠ 1, quindi normalizza peso sarà divisione del G.C.D risultante per i valori di peso.**

Nota: Quando G.C.D è uguale a 1, viene utilizzato il metodo precedente, altrimenti se G.C.D ≠ 1, quindi normalizza peso sarà divisione del G.C.D risultante per i valori di peso.

## **Manipolazione dei rapporti peso/carico CEF**

In alcuni scenari è possibile determinare il valore metrico del percorso da configurare per ottenere una distribuzione peso/carico risultante. È possibile determinare la metrica del percorso appropriata modificando il costo dei collegamenti e in base a fino a raggiungere o approssimare il valore richiesto. Si noti che non tutti i pesi che potrebbero essere necessari sono esattamente possibili, ma possiamo approssimare la distribuzione richiesta.

Prima di continuare, tenere in considerazione le restrizioni seguenti:

- a) Non tutte le distribuzioni peso/carico sono esattamente possibili, ma possiamo fare un'approssimazione.
- b.) Non superare mai i limiti del bucket di hash. - Ciò significa che la somma di tutti i pesi del percorso non può superare i bucket di hash; in tal caso, il peso deve essere normalizzato. Significa che, quando sommiamo tutti i pesi, non superiamo il limite dell'hash bucket.
- c.) ASR 9000 e CRS-X hanno un limite di 64 bucket di hash, XRv9000 un limite di 32 bucket di hash.
- d.) Quando si utilizza una versione precedente alla 6.4.1, la distribuzione del peso è diversa e

il percorso con il peso minore è sempre impostato su 1, mentre gli altri percorsi sono multipli di questo percorso, il che significa che può essere superiore a 1.

## Esempio 1: Rapporto peso/carico 26/5

Seguendo la stessa topologia in precedenza, desideriamo avere una distribuzione di peso 26/5 tra i due collegamenti.

i.) Inizialmente, i costi vengono impostati in modo uguale su tutti i percorsi ( $100 + 100 + 1 = 201$ ).

ii) Se si imposta la varianza UCMP sul valore massimo, verranno considerati tutti gli hop successivi.

iii) Se si controlla RIB, è possibile verificare lo stato predefinito in cui XR1 esegue ECMP.

```
RP/0/RP0/CPU0:XR1#show cef 3.3.3.3/32 detail
```

```
3.3.3.3/32, version 27, internal 0x1000001 0x0 (ptr 0xd3ecb50) [1], 0x0 (0xd583610), 0x0 (0x0)
Updated Nov 11 23:08:25.290
remote adjacency to GigabitEthernet0/0/0/0.12
Prefix Len 32, traffic index 0, precedence n/a, priority 1
gateway array (0xd416218) reference count 2, flags 0x0, source rib (7), 0 backups
    [3 type 3 flags 0x8401 (0xd4bc6f8) ext 0x0 (0x0)]
LW-LDI[type=3, refc=1, ptr=0xd583610, sh-ldi=0xd4bc6f8]
gateway array update type-time 1 Nov 11 23:08:25.290
LDI Update time Nov 11 23:08:25.297
LW-LDI-TS Nov 11 23:08:25.297
  via 12.0.0.2/32, GigabitEthernet0/0/0/0.12, 4 dependencies, weight 4294967295, class 0 [flags
0x0]
    path-idx 0 NHID 0x0 [0xe14b1b0 0x0]
    next hop 12.0.0.2/32
    remote adjacency
  via 13.0.0.2/32, GigabitEthernet0/0/0/0.13, 4 dependencies, weight 4294967295, class 0 [flags
0x0]
    path-idx 1 NHID 0x0 [0xe14b128 0x0]
    next hop 13.0.0.2/32
    remote adjacency

Weight distribution:
slot 0, weight 4294967295, normalized_weight 1, class 0
slot 1, weight 4294967295, normalized_weight 1, class 0
```

```
Load distribution: 0 1 (refcount 3)
```

Hash	OK	Interface	Address
0	Y	GigabitEthernet0/0/0/0.12	remote
1	Y	GigabitEthernet0/0/0/0.13	remote

In questo esempio verrà utilizzato un caso in cui si desidera impostare i seguenti pesi:

W1 = 26 (miglior costo primario)

W2 = 5 (miglior costo secondario)

È necessario prendere un percorso della gamba, per questo percorso, il costo dovrebbe già essere noto, in questo caso il percorso di riferimento sarà il percorso via Gi0/0/0/0.12. Il percorso della gamba sarà pre-calcolato con il costo da fine a fine, le metriche del percorso e il peso

richiesto per questo percorso sono:

i.)  $X1+Y1+D1 = 100 + 100 + 1 = 201$ . (Notare le variabili associate a ciascun collegamento nella topologia).

ii) Peso 1 = 26

iii) Peso 2 = 5

iv.) pm1 = 201 (percorso della gamba primaria); Peso = 26

v.) pm2 = ancora sconosciuto (percorso secondario); Peso = 5

Calcolo dei pesi.

Metrica percorso di pm2:  $pm2 = (26/5) * 201 = 1045$

Determinazione del costo del collegamento X2 su XR1.

$X2 = pm2 - (x2 + y1 + d1)$

$1045 - (100 + 100 + 1) = 844$

Configurazione del costo OSPF sul collegamento X2.

```
router ospf 1
  ucmp variance 10000
  area 0
  !
  interface GigabitEthernet0/0/0/0.13
    cost 844
```

Verificando la distribuzione peso/carico è possibile verificare che i pesi richiesti sono stati assegnati correttamente in CEF come previsto nei calcoli.

```
RP/0/RP0/CPU0:XR1#show cef 3.3.3.3/32 detail
```

```
3.3.3.3/32, version 37, internal 0x1000001 0x0 (ptr 0xd3ecce0) [1], 0x0 (0xd5835d8), 0x0 (0x0)
Updated Nov 11 23:17:47.945
remote adjacency to GigabitEthernet0/0/0/0.12
Prefix Len 32, traffic index 0, precedence n/a, priority 1
gateway array (0xd4163d8) reference count 1, flags 0x0, source rib (7), 0 backups
  [2 type 3 flags 0x8401 (0xd4bc7b8) ext 0x0 (0x0)]
LW-LDI[type=3, refc=1, ptr=0xd5835d8, sh-ldi=0xd4bc7b8]
gateway array update type-time 1 Nov 11 23:17:47.945
LDI Update time Nov 11 23:17:47.956
LW-LDI-TS Nov 11 23:17:47.956
  via 12.0.0.2/32, GigabitEthernet0/0/0/0.12, 6 dependencies, weight 4294967295, class 0 [flags 0x0]
    path-idx 0 NHID 0x0 [0xe14b1b0 0x0]
    next hop 12.0.0.2/32
    remote adjacency
  via 13.0.0.2/32, GigabitEthernet0/0/0/0.13, 6 dependencies, weight 913532538, class 0 [flags 0x0]
    path-idx 1 NHID 0x0 [0xe14b128 0x0]
    next hop 13.0.0.2/32
```





6	Y	GigabitEthernet0/0/0/0.12	remote
7	Y	GigabitEthernet0/0/0/0.12	remote
8	Y	GigabitEthernet0/0/0/0.12	remote
9	Y	GigabitEthernet0/0/0/0.12	remote
10	Y	GigabitEthernet0/0/0/0.12	remote
11	Y	GigabitEthernet0/0/0/0.12	remote
12	Y	GigabitEthernet0/0/0/0.12	remote
13	Y	GigabitEthernet0/0/0/0.12	remote
14	Y	GigabitEthernet0/0/0/0.12	remote
15	Y	GigabitEthernet0/0/0/0.12	remote
16	Y	GigabitEthernet0/0/0/0.12	remote
17	Y	GigabitEthernet0/0/0/0.12	remote
18	Y	GigabitEthernet0/0/0/0.12	remote
19	Y	GigabitEthernet0/0/0/0.12	remote
20	Y	GigabitEthernet0/0/0/0.12	remote
21	Y	GigabitEthernet0/0/0/0.12	remote
22	Y	GigabitEthernet0/0/0/0.12	remote
23	Y	GigabitEthernet0/0/0/0.12	remote
24	Y	GigabitEthernet0/0/0/0.12	remote
25	Y	GigabitEthernet0/0/0/0.12	remote
26	Y	GigabitEthernet0/0/0/0.12	remote
27	Y	GigabitEthernet0/0/0/0.12	remote
28	Y	GigabitEthernet0/0/0/0.12	remote
29	Y	GigabitEthernet0/0/0/0.12	remote
30	Y	GigabitEthernet0/0/0/0.13	remote