

# Een MPLS VPN via TE-tunnels uitvoeren

## Inhoud

[Inleiding](#)

[Voorwaarden](#)

[Vereisten](#)

[Gebruikte componenten](#)

[Conventies](#)

[Achtergrondinformatie](#)

[Eerste VPN-instelling tussen CE1 en CE2 zonder TE-tunnelbouw](#)

[Topologie](#)

[Configuratie](#)

[Verificatie](#)

[Zaak 1: VPN via een TE-tunnel wanneer de TE-tunnelleiding van PE1 naar PE2 is](#)

[Topologie](#)

[Configuratie](#)

[Verificatie](#)

[Zaak 2: VPN via een TE-tunnel wanneer de TE-tunnelleiding van PE1 naar P2 is](#)

[Topologie](#)

[Configuratie](#)

[Verificatie](#)

[verklaring](#)

[Oplossing](#)

[Zaak 3: VPN tussen CE1 en CE2 via een TE-tunnelverbinding van P1 tot P2 wanneer TDP/LDP niet is ingeschakeld](#)

[Topologie](#)

[Configuratie](#)

[Verificatie](#)

[Oplossing](#)

[Zaak 4: VPN via TE-tunnelverbinding tussen P1 en P2 met LDP ingeschakeld](#)

[Topologie](#)

[Configuratie](#)

[Verificatie](#)

[Zaak 5: MPLS VPN via een tunnelverbinding tussen P1 en PE2](#)

[Topologie](#)

[Configuratie](#)

[Verificatie](#)

[Bekende problemen](#)

[Conclusie](#)

[Gerelateerde informatie](#)

## Inleiding

Dit document biedt voorbeeldconfiguraties voor het implementeren van een Multiprotocol Label Switching (MPLS) VPN-tunnels via Traffic Engineering (TE) in een MPLS-netwerk. Om de voordelen van een MPLS VPN over TE-tunnels te behalen, moeten beide in het netwerk naast elkaar bestaan. Dit document illustreert verschillende scenario's die verklaren waarom het verzenden van pakketten binnen een MPLS VPN over TE-tunnels zou kunnen mislukken. Het biedt ook een mogelijke oplossing.

## Voorwaarden

### Vereisten

Lezers van dit document zouden kennis moeten hebben van deze onderwerpen:

- [MPLS traffic engineering en verbeteringen](#)
- [Een basis-MPLS VPN configureren](#)

### Gebruikte componenten

Dit document is niet beperkt tot specifieke software- en hardware-versies.

### Conventies

Raadpleeg [Cisco Technical Tips Conventions \(Conventies voor technische tips van Cisco\) voor meer informatie over documentconventies.](#)

### Achtergrondinformatie



Zoals in deze topologie wordt getoond, leert een eenvoudige MPLS VPN-configuratie, Provider Edge 1 (PE1) het VPN-label (Label 1 [L1]) voor VPN-prefix 172.16.13.0/24 via Multiprotocol Border Gateway Protocol (MPBGP) rechtstreeks van PE2, met de volgende hop als het PE2-loopback-adres. PE1 leert ook het etiket (L2) voor het PE2 loopback adres via het Protocol van de Distributie van het Etiket (LDP) van zijn volgende hop P1.

Wanneer het verzenden van gegevens naar het VPN prefix 172.16.13.13, gebruikt PE1 een etiket stapel {L2 L1} met L2 als buitenlabel. L2 wordt aangezwenkt door de router voor doorvoerlabel switch (LSR), P1. P2 plaatst de router L2 en stuurt het pakket door naar PE2 met slechts één L1. Om beter te begrijpen waarom P2 L2 pop heeft, zie rubriek 3.16 over voorlaatste hoppopping (PHP) in [RFC 3031](#). IP-pakketten naar het voorvoegsel 172.16.13.0/24 van VPN, versie 4 (IPv4), worden dus via een MPLS-netwerk geschakeld.

De MPLS VPN die operatie uitvoert faalt als een IP-router het pakket met L1 (VPN-label) ontvangt

als het enige routerlabel in plaats van de {L2 L1} labelstack. Dit komt voor omdat geen van de IP-routers L1 in zijn etiketterings- of informatiebasis (LFIB) heeft om het pakket te switchen.

Een MPLS TE gebruikt Resource Reservation Protocol (RSVP) om labels uit te wisselen. Wanneer een router voor zowel TE als TDP (TDP)/LDP (Tag Distribution Protocol) is ingesteld, ontvangt de router verschillende labels van zowel LDP als RSVP voor een bepaald voorvoegsel. De labels met LDP en RSVP hoeven niet in alle situaties hetzelfde te zijn. De router installeert een LDP-label in de verzendlijst als het voorvoegsel via een LDP-interface wordt geleerd, en het installeert het RSVP-label in de verzendlijst als het voorvoegsel via een TE-tunnelinterface wordt geleerd.

In het geval van een gewone TE-tunnel (zonder LDP/TDP dat in de tunnel is ingeschakeld) gebruikt de ingress LSR (de LSR aan het uiteinde van de TE-tunnel) hetzelfde label als wordt gebruikt om het uiteinde van de TE-tunnel te bereiken voor alle routes die door een TE-tunnel zijn geleerd.

Er is bijvoorbeeld een TE-tunnel van PE1 tot P2 die het voorvoegsel 10.11.11.11/32 over de tunnel leert. Het tunneleinde op P2 is 10.5.5.5, en het label om 10.5.5.5 in PE1 te bereiken is L3. PE1 gebruikt dan L3 om de bestemming 10.11.11.11/32 te bereiken, geleerd door de TE-tunnel.

In het [bovenstaande](#) scenario, wanneer er een TE-tunnel tussen PE1 en P2 is, moet u ervan uitgaan dat PE1 gegevens doorstuurt naar Customer Edge 2 (CE2). Als L4 het etiket van VPN is, door PE1 worden de gegevens met de etiketstapel {L3 L4} door gespoeld. P1 pop L3, en P2 ontvangt het pakket met L4. PE2 is de enige LSR die het pakket met het buitenlabel L4 correct kan doorsturen. P2 heeft geen MPBGP sessie met PE2, dus ontvangt het de L4 niet van PE2. Daarom heeft P2 geen kennis van L2 en laat het pakje vallen.

De configuraties en **tonen** resultaten die volgen en illustreren een mogelijke oplossing voor dit probleem.

## Eerste VPN-instelling tussen CE1 en CE2 zonder TE-tunnelbouw

### Topologie



### Configuratie

Hier zijn alleen de relevante onderdelen van de configuratiebestanden opgenomen:

```
PE1
hostname PE1
ip cef
!
ip vrf aqua
rd 100:1
route-target export 1:1
```

```
route-target import 1:1
!
mpls traffic-eng tunnels
!
interface Loopback0
 ip address 10.2.2.2 255.255.255.255
 no ip directed-broadcast
!
interface Ethernet2/0/1
 ip vrf forwarding aqua
 ip address 172.16.1.2 255.255.255.0
!
interface Ethernet2/0/2
 ip address 10.7.7.2 255.255.255.0
 ip router isis
 mpls traffic-eng tunnels
 tag-switching ip
!
router isis
 passive-interface Loopback0
 net 47.1234.2222.2222.00
 is-type level-1
 metric-style wide
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng level-1
!
router bgp 1
 bgp log-neighbor-changes
 neighbor 10.11.11.11 remote-as 1
 neighbor 10.11.11.11 update-source Loopback0
!
 address-family vpnv4
 neighbor 10.11.11.11 activate
 neighbor 10.11.11.11 send-community extended
 exit-address-family
!
 address-family ipv4
 neighbor 10.11.11.11 activate
 no auto-summary
 no synchronization
 exit-address-family
!
 address-family ipv4 vrf aqua
 redistribute connected
 no auto-summary
 no synchronization
 exit-address-family
```

## PE2

```
hostname PE2
!
ip vrf aqua
 rd 100:1
 route-target export 1:1
 route-target import 1:1
!
mpls traffic-eng tunnels
!
interface Loopback0
 ip address 10.11.11.11 255.255.255.255
!
interface POS0/1
```

```

ip address 10.12.12.10 255.255.255.0
ip router isis
mpls traffic-eng tunnels
tag-switching ip
crc 16
clock source internal
!
interface POS5/1
ip vrf forwarding aqua
ip address 172.16.13.11 255.255.255.0
crc 32
clock source internal
!
router isis
passive-interface Loopback0
mpls traffic-eng router-id Loopback0
mpls traffic-eng level-1
net 47.1234.1010.1010.1010.00
is-type level-1
metric-style wide
!
router bgp 1
bgp log-neighbor-changes
neighbor 10.2.2.2 remote-as 1
neighbor 10.2.2.2 update-source Loopback0
no auto-summary
!
address-family vpnv4
neighbor 10.2.2.2 activate
neighbor 10.2.2.2 send-community extended
exit-address-family
!
address-family ipv4 vrf aqua
redistribute connected
no auto-summary
no synchronization
exit-address-family
!

```

## Verificatie

PE2 leert PE1 VPN IPv4 prefix 172.16.1.0/24 over MPBGP piek tussen PE1 en PE2. Dit wordt hier getoond:

```
PE2# show ip route vrf aqua
```

```

Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR

```

```
Gateway of last resort is not set
```

```
10.0.0.0/24 is subnetted, 2 subnets
```

```
B 172.16.1.0 [200/0] via 10.2.2.2, 16:09:10
```

```
C 172.16.13.0 is directly connected, POS5/1
```

Op dezelfde manier leert PE1 PE2 VPN IPv4 prefix 172.16.13.0/24 over MPBGP tussen PE1 en PE2. Dit wordt hier getoond:

```
PE1# show ip route vrf aqua
```

```
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
```

```
Gateway of last resort is not set
```

```
10.0.0.0/24 is subnetted, 2 subnets
```

```
B 172.16.13.0 [200/0] via 10.11.11.11, 16:09:49
C 172.16.1.0 is directly connected, Ethernet2/0/1
```

```
PE1# show ip route vrf aqua 172.16.13.13
```

```
Routing entry for 172.16.13.0/24
  Known via "bgp 1", distance 200, metric 0, type internal
  Last update from 10.11.11.11 16:13:19 ago
  Routing Descriptor Blocks:
  * 10.11.11.11 (Default-IP-Routing-Table), from 10.11.11.11, 16:13:19 ago
    Route metric is 0, traffic share count is 1
    AS Hops 0, BGP network version 0
```

```
PE1# show ip cef vrf aqua 172.16.13.13
```

```
172.16.13.0/24, version 11, cached adjacency 10.7.7.7
0 packets, 0 bytes
  tag information set
    local tag: VPN route head
    fast tag rewrite with Et2/0/2, 10.7.7.7, tags imposed {17 12308}
  via 10.11.11.11, 0 dependencies, recursive
    next hop 10.7.7.7, Ethernet2/0/2 via 10.11.11.11/32
    valid cached adjacency
    tag rewrite with Et2/0/2, 10.7.7.7, tags imposed {17 12308}
!--- The label stack used to reach 172.16.13.13 is !--- {17 12308}, where 17 is the outer label to reach next hop 10.11.11.11 !--- and 12308 is the VPN IPv4 label for 172.16.13.0/24. PE1# show ip cef 10.11.11.11
```

```
10.11.11.11/32, version 31, cached adjacency 10.7.7.7
0 packets, 0 bytes
  tag information set
    local tag: 21
    fast tag rewrite with Et2/0/2, 10.7.7.7, tags imposed {17}
  via 10.7.7.7, Ethernet2/0/2, 1 dependency
    next hop 10.7.7.7, Ethernet2/0/2
    valid cached adjacency
    tag rewrite with Et2/0/2, 10.7.7.7, tags imposed {17}
!--- Outer label 17 is used to reach next hop 10.11.11.11.
```

CE1 kan dus 172.16.13.13 op het CE2-netwerk bereiken via de VPN-routing en -expediteur (VRF), die op PE1 is geconfigureerd met behulp van de labelstack {17 12308}, zoals hierboven wordt getoond.

Deze ping-uitvoer bevestigt de connectiviteit:

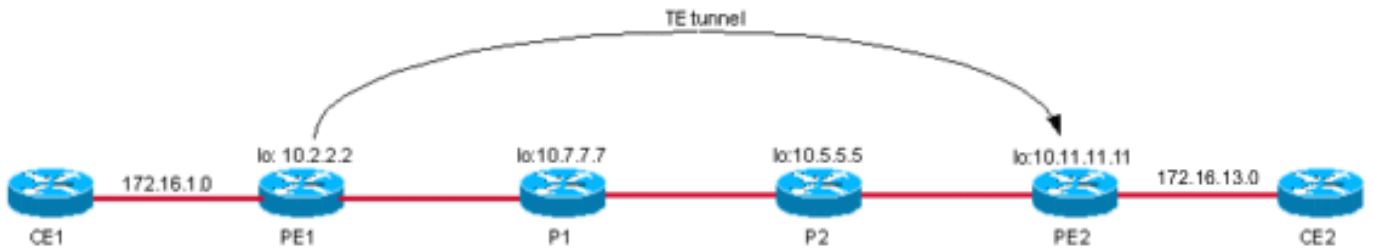
```
CE1# ping 172.16.13.13
```

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.13.13, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
```

[\*\*Zaak 1: VPN via een TE-tunnel wanneer de TE-tunnelleiding van\*\*](#)

# PE1 naar PE2 is

## Topologie



Wanneer de TE-tunnel tussen de PE-routers is gebouwd met automatische aankondiging, is de stap-PE BGP volgende hop bereikbaar via de TE-tunnelinterface. Daarom gebruikt PE1 het TE-label om PE2 te bereiken.

**Opmerking:** MPLS TE is onafhankelijk van LDP, wat betekent dat, als u een volledig netwerk van tunnels van PE tot PE hebt, u LDP in de routers effectief kunt uitschakelen en LDP niet op de TE-tunnelinterfaces hoeft te draaien. U moet echter alle tunnels naar de BGP volgende hop van de VPN versie 4 (VPNv4) routes bouwen. In het voorbeeld in deze [Configuration](#), zie je dat deze BGP volgende hop de Loopback0 op PE2 is, 10.11.11.11. Deze zelfde loopback is ook de tunnelbestemming voor de tunnel van PE1 naar PE2. Dit verklaart waarom, in dit voorbeeld, als er ook een tunnel is van PE2 naar PE1 voor het retourverkeer, je LDP kan uitschakelen kern. Vervolgens werkt het verzenden van CE naar CE met al VPNv4-verkeer dat over de TE-tunnels wordt vervoerd. Als de BGP volgende hop niet hetzelfde is als de TE tunnelbestemming, moet LDP in de kern en op de TE-tunnel worden uitgevoerd.

## Configuratie

De extra configuratie op PE1 voor de totstandbrenging van een PE-tunnel wordt hier getoond:

```
PE1

PE1# show run interface tunnel 0
!
interface Tunnel0
 ip unnumbered Loopback0
 no ip directed-broadcast
 no ip route-cache distributed
 tunnel destination 10.11.11.11
 tunnel mode mpls traffic-eng
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng path-option 10 dynamic
end
```

## Verificatie

```
PE1# show ip cef vrf aqua 172.16.13.13
172.16.13.0/24, version 11
0 packets, 0 bytes
```

```

tag information set
  local tag: VPN route head
  fast tag rewrite with Tu0, point2point, tags imposed {19 12308}
via 10.11.11.11, 0 dependencies, recursive
  next hop 10.11.11.11, Tunnel0 via 10.11.11.11/32
  valid adjacency
  tag rewrite with Tu0, point2point, tags imposed {19 12308}
!--- The label stack to reach 172.16.13.13 is {19 12308}. !--- BGP next hop for the VPNv4 prefix
is 10.11.11.11, which is !--- the same as the TE tunnel destination. PE1# show ip route
10.11.11.11
Routing entry for 10.11.11.11/32
  Known via "isis", distance 115, metric 40, type level-1
  Redistributing via isis
  Last update from 10.11.11.11 on Tunnel0, 00:02:09 ago
  Routing Descriptor Blocks:
  * 10.11.11.11, from 10.11.11.11, via Tunnel0
!--- The route is via Tunnel0. Route metric is 40, traffic share count is 1

```

Bevestig nu het etiket dat gebruikt wordt om de volgende hop 10.11.11.11 via Tunnel0 te bereiken.

```
PE1# show mpls traffic-eng tunnels tunnel 0
```

```

Name: PE1_t0 (Tunnel0) Destination: 10.11.11.11
Status:
  Admin: up      Oper: up      Path: valid      Signalling: connected

  path option 10, type dynamic (Basis for Setup, path weight 30)

Config Parameters:
  Bandwidth: 0 kbps (Global) Priority: 7 7 Affinity: 0x0/0xFFFF
  Metric Type: TE (default)
  AutoRoute: enabled LockDown: disabled Loadshare: 0 bw-based
  auto-bw: disabled

```

```

InLabel : -
OutLabel : Ethernet2/0/2, 19
!--- Label 19 from RSVP is used to reach destination 10.11.11.11/32. RSVP Signalling Info: Src
10.2.2.2, Dst 10.11.11.11, Tun_Id 0, Tun_Instance 31 RSVP Path Info: My Address: 10.7.7.2
Explicit Route: 10.7.7.7 10.8.8.7 10.8.8.5 10.12.12.10 10.11.11.11 Record Route: NONE Tspec: ave
rate=0 kbits, burst=1000 bytes, peak rate=0 kbits RSVP Resv Info: Record Route: NONE Fspec: ave
rate=0 kbits, burst=1000 bytes, peak rate=Inf Shortest Unconstrained Path Info: Path Weight: 30
(TE) Explicit Route: 10.7.7.2 10.7.7.7 10.8.8.7 10.8.8.5 10.12.12.10 10.11.11.11 History:
Tunnel: Time since created: 17 hours, 17 minutes Time since path change: 32 minutes, 54 seconds
Current LSP: Uptime: 32 minutes, 54 seconds Prior LSP: ID: path option 10 [14] Removal Trigger:
tunnel shutdown

```

Een andere manier om deze informatie snel te bekijken is door de output modifier in de opdrachten **show** te gebruiken, zoals hier wordt getoond:

```

PE1# show mpls traffic-eng tunnels tunnel 0 | include Label
  InLabel : -
  OutLabel : Ethernet2/0/2, 19
!--- This is the label to reach 10.11.11.11.

```

Kijk naar de stapel. Het is 19, dat het etiket TE is, gebruikt om pakketten naar volgende hop 10.11.11.0 over Tunnel0 te verzenden.

```

PE1# show tag forwarding-table 10.11.11.11 detail
Local  Outgoing  Prefix  Bytes tag  Outgoing  Next Hop
tag    tag or VC  or Tunnel Id  switched  interface

```



```

21      Pop tag      10.11.11.11/32    0          Tu0          point2point
      MAC/Encaps=14/18, MTU=1500, Tag Stack{19}, via Et2/0/2
      00603E2B02410060835887428847 00013000
      No output feature configured
      Per-packet load-sharing, slots: 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
PE1#

```

Dus, PE1 verstuurt een pakje voorbestemd voor 172.16.13.13 met de labelstack {19 12308}. P1 ruilt het etiket 19. Het pakje bereikt P2, wat het etiket van de buitenverpakking opvouwt. Daarna wordt het pakje naar PE2 doorgestuurd met alleen het label 12308.

Op PE2 wordt de verpakking met het label 12308 ontvangen en geschakeld volgens de informatie in de verzendingstabel. Dit wordt hier getoond:

```

PE2# show tag for tags 12308 detail
Local   Outgoing   Prefix          Bytes tag   Outgoing   Next Hop
tag     tag or VC  or Tunnel Id    switched   interface
12308   Aggregate  172.16.13.0/24[V] 12256
      MAC/Encaps=0/0, MTU=0, Tag Stack{}
      VPN route: aqua
      No output feature configured
      Per-packet load-sharing, slots: 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
PE2#

```

**Opmerking:** Er wordt geen uitgaande interface weergegeven omdat de afgaande tag Aggregate is. Dit komt doordat het prefix dat bij het label is gekoppeld de direct verbonden route is.

Pings van CE1 aan een gastheer op CE2 bevestigen de VPN connectiviteit over de TE tunnel:

```

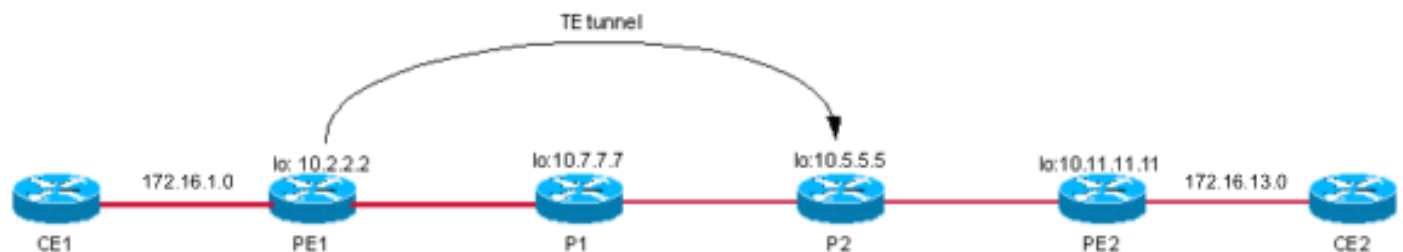
CE1# ping 172.16.13.13

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.13.13, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/13/36 ms
CE1#

```

## Zaak 2: VPN via een TE-tunnel wanneer de TE-tunnelleiding van PE1 naar P2 is

### Topologie



### Configuratie

De extra TE-configuratie over de basisconfiguratie op PE1 wordt hier weergegeven:

## PE1

```
PE1# show run interface tunnel 0
!
interface Tunnel0
 ip unnumbered Loopback0
 no ip directed-broadcast
 no ip route-cache distributed
 tunnel destination 10.5.5.5
 tunnel mode mpls traffic-eng
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng path-option 10 dynamic
end
!
```

## Verificatie

Controleer de route naar voorvoegsel 172.16.13.13 op PE1 VRF-water. Het wijst op de volgende hop 10.11.11.11/32 (over Tunnel0) die etiket stapel {19 12308} gebruikt.

```
PE1# show ip cef vrf aqua 172.16.13.13
172.16.13.0/24, version 11
0 packets, 0 bytes
 tag information set
   local tag: VPN route head
   fast tag rewrite with Tu0, point2point, tags imposed {19 12308}
 via 10.11.11.11, 0 dependencies, recursive
   next hop 10.5.5.5, Tunnel0 via 10.11.11.11/32
   valid adjacency
   tag rewrite with Tu0, point2point, tags imposed {19 12308}
PE1#
```

Etiket 19, het etiket op de buitenverpakking, wordt gebruikt om de volgende hop 10.11.11.11/32 te bereiken, zoals hier wordt getoond:

```
PE1# show ip cef 10.11.11.11
10.11.11.11/32, version 37
0 packets, 0 bytes
 tag information set
   local tag: 21
   fast tag rewrite with Tu0, point2point, tags imposed {19}
 via 10.5.5.5, Tunnel0, 1 dependency
   next hop 10.5.5.5, Tunnel0
   valid adjacency
   tag rewrite with Tu0, point2point, tags imposed {19}
```

```
PE1# show mpls traffic-eng tunnels tunnel 0
```

```
Name: PE1_t0 (Tunnel0) Destination: 10.5.5.5
Status:
  Admin: up      Oper: up      Path: valid      Signalling: connected

  path option 10, type dynamic (Basis for Setup, path weight 20)
```

```
Config Parameters:
```

```
Bandwidth: 0 kbps (Global) Priority: 7 7 Affinity: 0x0/0xFFFF
Metric Type: TE (default)
AutoRoute: enabled LockDown: disabled Loadshare: 0 bw-based
auto-bw: disabled
```

```

InLabel : -
OutLabel : Ethernet2/0/2, 19
RSVP Signalling Info:
  Src 10.2.2.2, Dst 10.5.5.5, Tun_Id 0, Tun_Instance 33
RSVP Path Info:
  My Address: 10.7.7.2
  Explicit Route: 10.7.7.7 10.8.8.7 10.8.8.5 10.5.5.5
  Record Route: NONE
  Tspec: ave rate=0 kbits, burst=1000 bytes, peak rate=0 kbits
RSVP Resv Info:
  Record Route: NONE
  Fspec: ave rate=0 kbits, burst=1000 bytes, peak rate=Inf
Shortest Unconstrained Path Info:
  Path Weight: 20 (TE)
  Explicit Route: 10.7.7.2 10.7.7.7 10.8.8.7 10.8.8.5
                  10.5.5.5
History:
  Tunnel:
    Time since created: 17 hours, 31 minutes
    Time since path change: 8 minutes, 49 seconds
  Current LSP:
    Uptime: 8 minutes, 49 seconds
    Selection: reoptimization
  Prior LSP:
    ID: path option 10 [31]
    Removal Trigger: path verification failed
PE1#

```

```
PE1# show mpls traffic-eng tunnels tunnel 0 | i Label
```

```

InLabel : -
OutLabel : Ethernet2/0/2, 19

```

```
PE1#
```

Het pakket van PE1 wordt verzonden over de TE tunnel met de labelstack {19 12308}. Zodra P1 het pakket ontvangt, verschijnt het (PHP) de tag 19 en stuurt het pakket met label stack {12308}. De show opdracht bevestigt dit:

```
P1> show tag for tag 19
```

Local tag	Outgoing tag or VC	Prefix or Tunnel Id	Bytes tag switched	Outgoing interface	Next Hop
19	Pop tag	10.2.2.2 0 [33]	2130	Et2/0	10.8.8.5

```
P1> show tag for tag 19 detail
```

Local tag	Outgoing tag or VC	Prefix or Tunnel Id	Bytes tag switched	Outgoing interface	Next Hop
19	Pop tag	10.2.2.2 0 [33]	2257	Et2/0	10.8.8.5

MAC/Encaps=14/14, MTU=1504, Tag Stack{  
006009E08B0300603E2B02408847  
No output feature configured

```
P1>
```

Wanneer P2 het pakket met de labelstack {12308} ontvangt, controleert het zijn LFIB en daalt het pakket omdat er geen overeenkomende producten zijn. Dit is de opdrachtoutput van de show op P2:

```
P2# show tag forwarding-table tags 12308 detail
```

Local tag	Outgoing tag or VC	Prefix or Tunnel Id	Bytes tag switched	Outgoing interface	Next Hop
-----------	--------------------	---------------------	--------------------	--------------------	----------

```
P2#
```

```
P2#
7w4d: TAG: Et0/3: recvd: CoS=0, TTL=253, Tag(s)=12308
7w4d: TAG: Et0/3: recvd: CoS=0, TTL=253, Tag(s)=12308
7w4d: TAG: Et0/3: recvd: CoS=0, TTL=253, Tag(s)=12308
7w4d: TAG: Et0/3: recvd: CoS=0, TTL=253, Tag(s)=12308
P2#
P2#
```

## verklaring

De oplossing voor dit probleem is om TDP/LDP op de TE-tunnel in te schakelen en er een tag-switched interface van te maken. In het voorbeeld in de [Oplossing](#) wordt getoond, wordt TDP geactiveerd op de Tunnel 0 van PE1. P2 wordt gevormd om gerichte hellos te accepteren en gerichte TDP burens te vormen. PE1 ontvangt het label voor 10.11.11.11 van P2 via LDP. Nu Tunnel0 een tag-switched interface heeft gemaakt en TDP is ingeschakeld voor het verkeer op 10.11.11.11, gebruikt PE1 beide etiketten; het gebruikt het RSVP-label om het doeleinde te bereiken en het TDP-label om 10.11.11.11 te bereiken.

In dit scenario gebruikt PE1 de etiket stapel {L2 L3 L1} om gegevens naar CE2 door te sturen als deze punten waar zijn:

- L1 is het VPN-label.
- L2 is het RSVP-label om het doeleinde te bereiken.
- L3 is het TDP-label om 10.11.11.11 te bereiken (ontvangen van P2).

## Oplossing

De oplossing is om TDP over de TE-tunnel toe te staan.

## Configuratie

Hier is de TE-tunnelconfiguratie op PE1 met een TDP ingeschakeld. De toevoegingen zijn onscherp.

```
PE1
PE1# show run interface tunnel 0
!
interface Tunnel0
 ip unnumbered Loopback0
 no ip directed-broadcast
 no ip route-cache distributed
 tag-switching ip
 !--- This enables TDP. tunnel destination 10.5.5.5
 tunnel mode mpls traffic-eng tunnel mpls traffic-eng
 autoroute announce tunnel mpls traffic-eng path-option
 10 dynamic end !
```

Dit is de extra configuratie aan het uiteinde van de TE-tunnel om gerichte TDP-hellos te accepteren:

```
P2# show run | i directed-hello
tag-switching tdp discovery directed-hello accept
```

```
!--- This configures P2 to accept directed TDP hellos. P2#
```

## Verificatie

```
PE1# show tag tdp neighbor | i Peer
```

```
Peer TDP Ident: 10.7.7.7:0; Local TDP Ident 10.2.2.2:0  
Peer TDP Ident: 10.5.5.5:0; Local TDP Ident 10.2.2.2:0
```

```
PE1#
```

```
PE1# show ip cef vrf aqua 172.16.13.13
```

```
172.16.13.0/24, version 11
```

```
0 packets, 0 bytes
```

```
tag information set
```

```
local tag: VPN route head
```

```
fast tag rewrite with Tu0, point2point, tags imposed {19 18 12308}  
via 10.11.11.11, 0 dependencies, recursive
```

```
next hop 10.5.5.5, Tunnel0 via 10.11.11.11/32
```

```
valid adjacency
```

```
tag rewrite with Tu0, point2point, tags imposed {19 18 12308}
```

```
PE1#
```

```
PE1# show mpls traffic-eng tunnels tunnel 0 | i Label
```

```
InLabel : -
```

```
OutLabel : Ethernet2/0/2, 19
```

```
!--- This is the TE label learned via RSVP. PE1# PE1# show tag tdp bind 10.11.11.11 32
```

```
tib entry: 10.11.11.11/32, rev 20
```

```
local binding: tag: 21
```

```
remote binding: tsr: 10.7.7.7:0, tag: 17
```

```
remote binding: tsr: 10.5.5.5:0, tag: 18
```

```
!--- This is the TDP label from P2.
```

Wanneer P1 het pakket met de labelstack {19 18 12308} ontvangt, verschijnt het de tag 19 en stuurt het pakket met de labelstack {18 12308} naar P2. P2 controleert zijn LFIB voor etiket 18, dan stelt het de tag op en stuurt het over de uitgaande interface 2/0/0 naar PE1. ontvangt het pakket met etiket 12308 en switch het met succes aan CE2.

```
P2# show tag for tag 18
```

Local tag	Outgoing tag or VC	Prefix or Tunnel Id	Bytes tag switched	Outgoing interface	Next Hop
18	Pop tag	10.11.11.11/32	117496	POS2/0/0	point2point

```
P2# show tag tdp discovery
```

```
Local TDP Identifier:
```

```
10.5.5.5:0
```

```
Discovery Sources:
```

```
Interfaces:
```

```
Ethernet0/3 (tdp): xmit/rcv
```

```
TDP Id: 10.7.7.7:0
```

```
POS2/0/0 (tdp): xmit/rcv
```

```
TDP Id: 10.11.11.11:0
```

```
Directed Hellos:
```

```
10.5.5.5 -> 10.2.2.2 (tdp): passive, xmit/rcv
```

```
TDP Id: 10.2.2.2:0
```

```
P2# show tag tdp neighbor 10.2.2.2
```

```
Peer TDP Ident: 10.2.2.2:0; Local TDP Ident 10.5.5.5:0
```

```
TCP connection: 10.2.2.2.711 - 10.5.5.5.11690
```

```
State: Oper; PIEs sent/rcvd: 469/465; Downstream
```

```
Up time: 01:41:08
```

```
TDP discovery sources:
```

```
Directed Hello 10.5.5.5 -> 10.2.2.2, passive
```

```
Addresses bound to peer TDP Ident:
10.7.7.2      172.16.47.166  10.2.2.2
```

```
PE1# show tag tdp neighbor 10.5.5.5
```

```
Peer TDP Ident: 10.5.5.5:0; Local TDP Ident 10.2.2.2:0
TCP connection: 10.5.5.5.11690 - 10.2.2.2.711
State: Oper; PIEs sent/rcvd: 438/441; Downstream
Up time: 01:35:08
TDP discovery sources:
```

```
Directed Hello 10.2.2.2 -> 10.5.5.5, active
```

```
!--- This indicates the directed neighbor. Addresses bound to peer TDP Ident: 10.5.5.5
10.12.12.5 10.8.8.5 PE1# show ip route 10.11.11.11
```

```
Routing entry for 10.11.11.11/32
```

```
Known via "isis", distance 115, metric 40, type level-1
```

```
Redistributing via isis
```

```
B Last update from 10.5.5.5 on Tunnel0, 01:52:21 ago
```

```
Routing Descriptor Blocks:
```

```
* 10.5.5.5, from 10.11.11.11, via Tunnel0
```

```
Route metric is 40, traffic share count is 1
```

Een ping opdracht van CE1 naar een host op CE2 bevestigt de oplossing.

```
CE1# ping 172.16.13.13
```

```
Type escape sequence to abort.
```

```
Sending 5, 100-byte ICMP Echos to 172.16.13.13, timeout is 2 seconds:
```

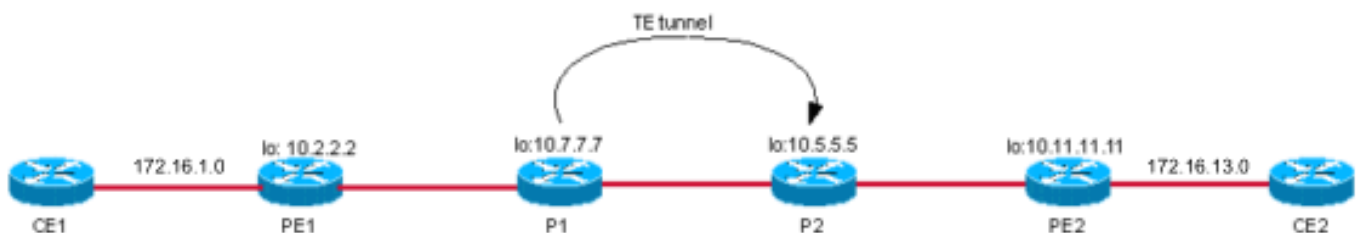
```
!!!!!
```

```
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/4/4 ms
```

```
CE1#
```

## Zaak 3: VPN tussen CE1 en CE2 via een TE-tunnelverbinding van P1 tot P2 wanneer TDP/LDP niet is ingeschakeld

### Topologie



### Configuratie

De tunnelconfiguratie op PE1 wordt hier getoond:

#### PE1

```
P1# show run interface tunnel 0
Building configuration...

Current configuration : 255 bytes
!
interface Tunnel0
 ip unnumbered Loopback0
```

```
no ip directed-broadcast
ip route-cache distributed
tunnel destination 10.5.5.5
tunnel mode mpls traffic-eng
tunnel mpls traffic-eng autoroute announce
tunnel mpls traffic-eng path-option 10 dynamic
end
```

## Verificatie

Controleer hoe pakketten bestemd voor CE2 172.16.13.13 hier worden geschakeld. De opdrachtoutput van **ip cef** toont dat pakketten naar bestemming 172.16.13.13 met de labelstack {17 12308} worden geschakeld:

```
PE1# show ip cef vrf aqua 172.16.13.13
172.16.13.0/24, version 18, cached adjacency 10.7.7.7
0 packets, 0 bytes
tag information set
  local tag: VPN route head
  fast tag rewrite with Et2/0/2, 10.7.7.7, tags imposed {17 12308}
via 10.11.11.11, 0 dependencies, recursive
  next hop 10.7.7.7, Ethernet2/0/2 via 10.11.11.11/32
  valid cached adjacency
  tag rewrite with Et2/0/2, 10.7.7.7, tags imposed {17 12308}
```

Wanneer P1 dit pakket ontvangt, verwijdert het het buitenlabel 17 en switch het pakket na het kijken in de IP-routingtabel naar Tunnel0. Let op het impliciete-ongeldige OutLabel in deze uitvoer; dit betekent dat de uittrekkende interface niet is ingeschakeld.

```
P1# show ip cef 10.11.11.11 detail
10.11.11.11/32, version 52
0 packets, 0 bytes
tag information set
  local tag: 17
  fast tag rewrite with Tu0, point2point, tags imposed {}
via 10.5.5.5, Tunnel0, 0 dependencies
  next hop 10.5.5.5, Tunnel0
  valid adjacency
  tag rewrite with Tu0, point2point, tags imposed {}
```

```
P1# show mpls traffic-eng tunnel tunnel 0 | i Label
InLabel : -
OutLabel : Ethernet2/0, implicit-null
```

```
P1# show tag for 10.11.11.11 detail
```

```
Local  Outgoing  Prefix          Bytes tag  Outgoing  Next Hop
tag    tag or VC  or Tunnel Id    switched   interface
17     Untagged  10.11.11.11/32  882        Tu0        point2point
      MAC/Encaps=14/14, MTU=1500, Tag Stack{ }, via Et2/0
      006009E08B0300603E2B02408847
      No output feature configured
      Per-packet load-sharing, slots: 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
```

```
P1# show ip route 10.11.11.11
```

```
Routing entry for 10.11.11.11/32
  Known via "isis", distance 115, metric 30, type level-1
  Redistributing via isis
  Last update from 10.5.5.5 on Tunnel0, 00:03:20 ago
  Routing Descriptor Blocks:
  * 10.5.5.5, from 10.11.11.11, via Tunnel0
```

Route metric is 30, traffic share count is 1

Zodra P2 het pakket met etiket 12308 ontvangt, bekijkt het zijn verzendingstabel. Omdat P2 op geen enkele manier vanaf CE2 op de hoogte kan zijn van de VPN-tag 12308, zet het pakje neer.

```
P2# show tag for tag 12308 detail
```

```
Local   Outgoing   Prefix           Bytes tag   Outgoing   Next Hop
tag     tag or VC   or Tunnel Id     switched    interface

```

Dit breekt het pad van VPN-pakketten die bestemd zijn voor CE2. Het wordt bevestigd door het pingelen aan CE2 172.16.13.13/32.

```
PE1#
CE1# ping 172.16.13.13
```

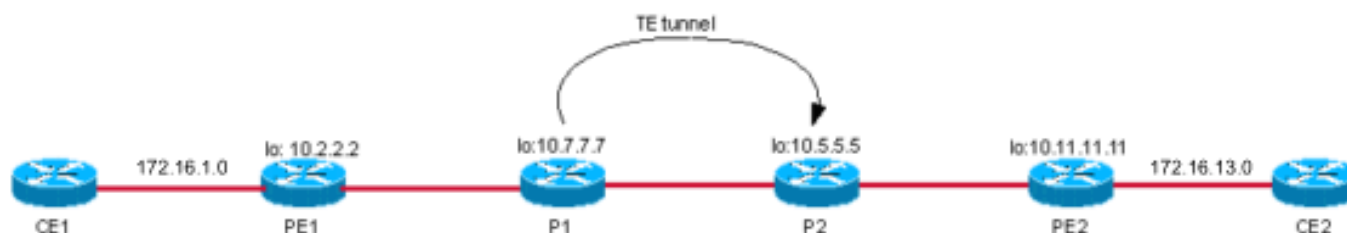
```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.13.13, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
CE1#
```

## Oplossing

De oplossing is om LDP/TDP via de tunnel in te schakelen. In de volgende paragraaf wordt deze oplossing besproken.

## Zaak 4: VPN via TE-tunnelverbinding tussen P1 en P2 met LDP ingeschakeld

### Topologie



### Configuratie

Als LDP is ingeschakeld in de tunnel, verschijnen de configuraties op P1 zoals hier wordt getoond. Toevoegingen zijn vet.

```
PE1
-----
P1# show run interface tunnel 0
Building configuration...

Current configuration : 273 bytes
!
interface Tunnel0
 ip unnumbered Loopback0
```



```

no ip directed-broadcast
ip route-cache distributed
mpls label protocol ldp
tunnel destination 10.5.5.5
tunnel mode mpls traffic-eng
tunnel mpls traffic-eng autoroute announce
tunnel mpls traffic-eng path-option 10 dynamic
end
!
```

## Verificatie

PE1 stuurt pakketten naar voorvoegsel 172.16.13.13/32 met de labelstack {17 12308}.

```

PE1#
PE1# show tag for 10.11.11.11 detail
Local   Outgoing   Prefix           Bytes tag  Outgoing   Next Hop
tag     tag or VC  or Tunnel Id     switched   interface
21      17         10.11.11.11/32   0          Et2/0/2    10.7.7.7
        MAC/Encaps=14/18, MTU=1500, Tag Stack{17}
        00603E2B02410060835887428847 00011000
        No output feature configured
        Per-packet load-sharing, slots: 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
```

```

PE1#
PE1# show ip cef 10.11.11.11 detail
10.11.11.11/32, version 60, cached adjacency 10.7.7.7
0 packets, 0 bytes
tag information set
  local tag: 21
  fast tag rewrite with Et2/0/2, 10.7.7.7, tags imposed {17}
via 10.7.7.7, Ethernet2/0/2, 1 dependency
  next hop 10.7.7.7, Ethernet2/0/2
  valid cached adjacency
  tag rewrite with Et2/0/2, 10.7.7.7, tags imposed {17}
```

```

PE1# show ip cef vrf aqua 172.16.13.13
172.16.13.0/24, version 18, cached adjacency 10.7.7.7
0 packets, 0 bytes
tag information set
  local tag: VPN route head
  fast tag rewrite with Et2/0/2, 10.7.7.7, tags imposed {17 12308}
via 10.11.11.11, 0 dependencies, recursive
  next hop 10.7.7.7, Ethernet2/0/2 via 10.11.11.11/32
  valid cached adjacency
  tag rewrite with Et2/0/2, 10.7.7.7, tags imposed {17 12308}
```

P1 ontvangt het pakket met label stack {17 12308} en kijkt naar zijn LFIB voor label 17.

```

P1# show tag for tag 17 detail
Local   Outgoing   Prefix           Bytes tag  Outgoing   Next Hop
tag     tag or VC  or Tunnel Id     switched   interface
17      18         10.11.11.11/32   1158      Tu0         point2point
        MAC/Encaps=14/18, MTU=1496, Tag Stack{18}, via Et2/0
        006009E08B0300603E2B02408847 00012000
        No output feature configured
        Per-packet load-sharing, slots: 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
P1#
```

```

P1# show ip cef 10.11.11.11 detail
10.11.11.11/32, version 52
0 packets, 0 bytes
tag information set
  local tag: 17
  fast tag rewrite with Tu0, point2point, tags imposed {18}
via 10.5.5.5, Tunnel0, 0 dependencies
  next hop 10.5.5.5, Tunnel0
  valid adjacency
  tag rewrite with Tu0, point2point, tags imposed {18}

```

Het toont dat etiket 17 moet worden omgewisseld om etiket 18 te etiketteren. Daarom is dat pakje over de tunnelinterface met de labelstack {18 12308} geschakeld.

P2 ontvangt het pakket over zijn tunnelinterface met labelstack {18 12308}. Deze optie bevat tag 18 (omdat het de voorlaatste hoprouter is) en switch het pakje naar PE2 met het label 12308.

```

P2# show tag for tag 18 detail
Local  Outgoing  Prefix          Bytes tag  Outgoing  Next Hop
tag    tag or VC  or Tunnel Id    switched   interface
18    Pop tag   10.11.11.11/32  127645    PO2/0/0   point2point
      MAC/Encaps=4/4, MTU=4474, Tag Stack{
      0F008847
      No output feature configured
      Per-packet load-sharing, slots: 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
P2#

```

PE2 ontvangt het pakket met etiket 12308, dat het pakket met succes aan CE2 switch.

```

PE2# show tag forwarding tags 12308 detail
Local  Outgoing  Prefix          Bytes tag  Outgoing  Next Hop
tag    tag or VC  or Tunnel Id    switched   interface
12308 Aggregate 172.16.13.0/24[V] 12256
      MAC/Encaps=0/0, MTU=0, Tag Stack{
      VPN route: aqua
      No output feature configured
      Per-packet load-sharing, slots: 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
PE2#

```

```

CE1# ping 172.16.13.13

```

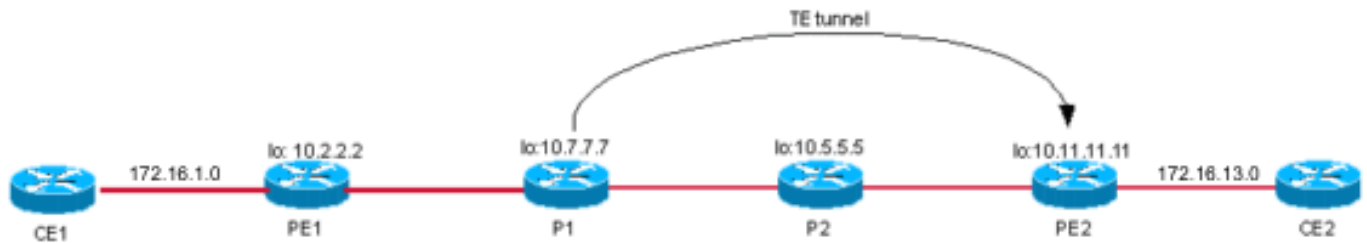
```

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.13.13, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
CE1#

```

## [Zaak 5: MPLS VPN via een tunnelverbinding tussen P1 en PE2](#)

### [Topologie](#)



## Configuratie

### PE1

```

P1# show run interface tunnel 0
Building configuration...

Current configuration : 258 bytes
!
interface Tunnel0
 ip unnumbered Loopback0
 no ip directed-broadcast
 ip route-cache distributed
 tunnel destination 10.11.11.11
 tunnel mode mpls traffic-eng
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng path-option 10 dynamic
end

```

## Verificatie

PE1 stuurt een pakje voorbestemd voor 172.16.13.13 naar de volgende hop 10.11.11.11 met de labelstack {17 12308}.

```

PE1# show ip cef vrf aqua 172.16.13.13
172.16.13.0/24, version 18, cached adjacency 10.7.7.7
0 packets, 0 bytes
 tag information set
   local tag: VPN route head
   fast tag rewrite with Et2/0/2, 10.7.7.7, tags imposed {17 12308}
 via 10.11.11.11, 0 dependencies, recursive
   next hop 10.7.7.7, Ethernet2/0/2 via 10.11.11.11/32
   valid cached adjacency
   tag rewrite with Et2/0/2, 10.7.7.7, tags imposed {17 12308}

```

P1 ontvangt het pakket met labelstack {17 12308} P1 kijkt naar zijn LFIB tabel en controleert de tag stack {17} en switches het pakket met label {17} naar P2.

```

P1# show tag for 10.11.11.11 detail
Local  Outgoing  Prefix          Bytes tag  Outgoing  Next Hop
tag    tag or VC  or Tunnel Id   switched  interface
17     Untagged  10.11.11.11/32  411       Tu0       point2point
      MAC/Encaps=14/18, MTU=1500, Tag Stack{17}, via Et2/0
      006009E08B0300603E2B02408847 00011000
      No output feature configured

```

Per-packet load-sharing, slots: 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

P1# **show tag for tag 17 detail**

```
Local  Outgoing  Prefix          Bytes tag  Outgoing  Next Hop
tag    tag or VC  or Tunnel Id    switched   interface
17     Untagged  10.11.11.11/32  685        Tu0        point2point
      MAC/Encaps=14/18, MTU=1500, Tag Stack{17}, via Et2/0
      006009E08B0300603E2B02408847 00011000
      No output feature configured
```

Per-packet load-sharing, slots: 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

P1#

P1# **show ip cef 10.11.11.11**

```
10.11.11.11/32, version 67
0 packets, 0 bytes
tag information set
local tag: 17
fast tag rewrite with Tu0, point2point, tags imposed {17}
via 10.11.11.11, Tunnel0, 0 dependencies
next hop 10.11.11.11, Tunnel0
valid adjacency
tag rewrite with Tu0, point2point, tags imposed {17}
```

P2 ontvangt het pakket met labelstack {17 12308}. P2, de voorlaatste hoprouter, pops label 17.

P2# **show tag for tag 17 detail**

```
Local  Outgoing  Prefix          Bytes tag  Outgoing  Next Hop
tag    tag or VC  or Tunnel Id    switched   interface
17     Pop tag  10.7.7.7 0 [5]  535        P02/0/0    point2point
      MAC/Encaps=4/4, MTU=4474, Tag Stack{}
      0F008847
      No output feature configured
```

P2#

PE2 ontvangt vervolgens de verpakking met het label 12308. P2 is zich ervan bewust dat de bestemming voor label 12308 rechtstreeks is verbonden. Daarom is het **pingelen** van CE1 tot CE2 10.

PE2# **show tag for tag 12308 detail**

```
Local  Outgoing  Prefix          Bytes tag  Outgoing  Next Hop
tag    tag or VC  or Tunnel Id    switched   interface
12308  Aggregate 172.16.13.0/24[V] 12776
      MAC/Encaps=0/0, MTU=0, Tag Stack{}
      VPN route: aqua
      No output feature configured
```

Per-packet load-sharing, slots: 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

PE2#

**Opmerking:** Er wordt geen uitgaande interface weergegeven omdat de afgaande tag `Aggregate` is. Dit komt doordat het prefix dat bij het label is gekoppeld de direct verbonden route is.

CE1# **ping 172.16.13.13**

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 172.16.13.13, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 4/4/4 ms

CE1#

## Bekende problemen

Raadpleeg de [melding uit het veld: MPLS VPN met TE en MPLS InterAS Advisory op Cisco IOS® Software](#) voor meer informatie.

## Conclusie

Wanneer de TE-tunnel op stress-PE wordt afgesloten, werken MPLS VPN en TE samen zonder extra configuratie. Wanneer de TE-tunnel op om het even welke routers (vóór PE in de kern) wordt beëindigd, zal het MPLS VPN-verkeer mislukken omdat pakketten met VPN-labels arriveren als de buitenlabels, die niet in LFIBs van deze apparaten staan. Daarom zijn deze intermediaire routers niet in staat pakketten naar de eindbestemming, het VPN-clientnetwerk, door te sturen. In zo een geval zou LDP/TDP op de TE-tunnel moeten worden ingeschakeld om het probleem op te lossen.

## Gerelateerde informatie

- [MPLS FAQ voor beginners](#)
- [Probleemoplossing voor MPLS VPN](#)
- [MPLS basis traffic engineering met OSPF-configuratievoorbeld](#)
- [Een basis-MPLS VPN configureren](#)
- [LSP-fouten voor probleemoplossing in MPLS VPN](#)
- [MPLS-ondersteuningspagina](#)
- [Technische ondersteuning - Cisco-systemen](#)