

# 整合MPLS組態範例

## 目錄

[簡介](#)

[必要條件](#)

[需求](#)

[採用元件](#)

[背景](#)

[架構](#)

[設定](#)

[驗證](#)

[疑難排解](#)

[相關資訊](#)

## 簡介

本檔案介紹整合多重協定標籤交換(MPLS)的用途，並提供組態範例。

## 必要條件

### 需求

本文件沒有特定需求。

### 採用元件

本文件所述內容不限於特定軟體和硬體版本。

本文中的資訊是根據特定實驗室環境內的裝置所建立。文中使用到的所有裝置皆從已清除（預設）的組態來啟動。如果您的網路正在作用，請確保您已瞭解任何指令可能造成的影響。

## 背景

統一MPLS的目的全部在於擴展。為了擴展MPLS網路（該網路的一部分有不同型別的平台和服務），將網路劃分為不同區域是很有意義的。典型的設計引入了一個分層結構，其核心位於中心，匯聚位於側面。為了進行擴展，與聚合相比，核心中可以有不同的內部網關協定(IGP)。為了擴展，您不能將IGP字首從一個IGP分配到另一個IGP。如果不將IGP字首從一個IGP分發到另一個IGP，則端

到端標籤交換路徑(LSP)將不可行。

為了端到端提供MPLS服務，您需要將LSP設定為端到端。目標是保持MPLS服務(MPLS VPN、MPLS L2VPN)不變，但引入更高的可擴充性。為此，請將某些IGP字首移動到邊界網關協定(BGP)(提供商邊緣(PE)路由器的環回字首)，然後由提供商邊緣將字首端到端分發。

## 架構

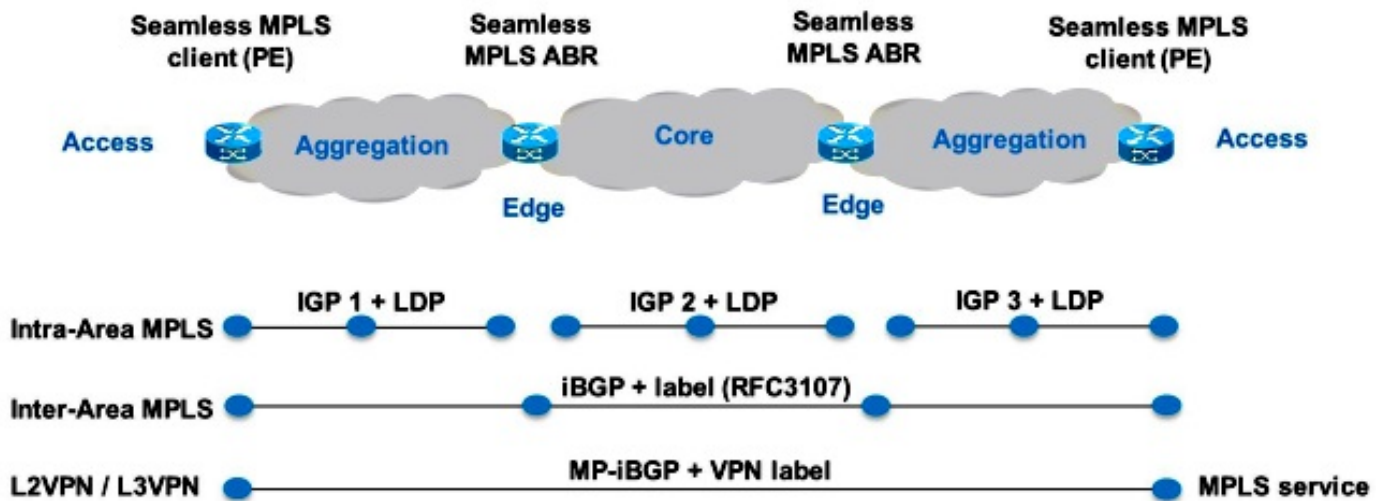


Figure 1

圖1顯示了具有三個不同區域的網路：一個核心和兩側的兩個聚合區域。每個區域運行自己的IGP，在區域邊界路由器(ABR)上它們之間不進行重新分配。需要使用BGP來提供端到端MPLS LSP。BGP在整個域內使用標籤通告PE路由器的環回，並提供端到端LSP。BGP是使用RFC 3107在PE和ABR之間部署，這表示BGP會傳送IPv4首碼+標籤(AFI/SAFI 1/4)。

由於網路的核心部分和聚合部分被整合並提供端到端LSP，因此Unified MPLS解決方案也稱為「無縫MPLS」。

此處未使用新技術或協定，僅使用MPLS、標籤發佈協定(LDP)、IGP和BGP。由於您不想將PE路由器的環回字首從網路的一部分分配到另一部分，因此需要在BGP中攜帶字首。內部邊界網關協定(iBGP)在一個網路中使用，因此字首的下一跳地址是PE路由器的環回字首，而網路其他部分的IGP不知道這一點。這意味著下一跳地址不能用於遞迴為IGP字首。訣竅是使ABR路由器成為路由反射器(RR)，並將下一跳設定為自己，即使對於反射的iBGP字首也是如此。為了讓這個起作用，需要一個新旋鈕。

只有RR需要更新的軟體來支援此體系結構。由於RR通告BGP字首並將下一跳設定為自身，因此它們將本地MPLS標籤分配給BGP字首。這表示在資料平面中，在這些端到端LSP上轉發的資料包在標籤堆疊中具有額外的MPLS標籤。RR位於轉發路徑中。

**附註：**在此架構上，提供任何MPLS服務。例如，在PE路由器之間提供MPLS VPN或MPLS L2VPN。這些資料包的資料平面的區別在於，它們現在標籤堆疊中有三個標籤，而當未使用

統一MPLS時，標籤堆疊中有兩個標籤。

有兩種可能情況：

- ABR不會將ABR向網路的匯聚部分通告的字首（由BGP反映）的下一跳設定為自身。因此，ABR需要將ABR的環回字首從核心IGP重新分配到聚合IGP。如果完成此操作，仍然可以擴展。只需將ABR環回字首（來自核心）通告到匯聚部分，而不需要將來自PE路由器的環回字首通告到遠端匯聚部分。
- ABR將ABR向聚合部分通告的字首（由BGP反映）的下一跳設定為self。因此，ABR不需要將ABR的環回字首從核心IGP重新分配到聚合IGP。

在這兩種情況下，ABR將ABR通告的（由BGP反映的）字首的下一跳設定為自身，從網路的聚合部分到核心部分。如果不這樣做，ABR需要將PE的環回字首從聚合IGP重新分配到核心IGP。如果這樣做，則沒有可擴充性。

若要將反射的iBGP路由的下一跳設定為self，必須配置**neighbor x.x.x.x next-hop-self all**命令。

## 設定

這是場景2中PE路由器和ABR的配置。

**附註：**該拓撲如圖2所示。示例服務是xconnect(MPLS L2VPN)。在PE路由器和ABR之間存在BGP for IPv4 +標籤。

### PE1

```
interface Loopback0
 ip address 10.100.1.4 255.255.255.255

!
interface Ethernet1/0
 no ip address
 xconnect 10.100.1.5 100 encapsulation mpls
!
router ospf 2
 network 10.2.0.0 0.0.255.255 area 0
 network 10.100.1.4 0.0.0.0 area 0
!
router bgp 1
 bgp log-neighbor-changes
 network 10.100.1.4 mask 255.255.255.255
 neighbor 10.100.1.1 remote-as 1
 neighbor 10.100.1.1 update-source Loopback0
 neighbor 10.100.1.1 send-label
```

### RR1

```
interface Loopback0
 ip address 10.100.1.1 255.255.255.255
router ospf 1
 network 10.1.0.0 0.0.255.255 area 0
 network 10.100.1.1 0.0.0.0 area 0
```

```
!  
router ospf 2  
  redistribute ospf 1 subnets match internal route-map ospf1-into-ospf2  
  network 10.2.0.0 0.0.255.255 area 0  
!  
router bgp 1  
  bgp log-neighbor-changes  
  neighbor 10.100.1.2 remote-as 1  
  neighbor 10.100.1.2 update-source Loopback0  
  neighbor 10.100.1.2 next-hop-self all  
  neighbor 10.100.1.2 send-label  
  neighbor 10.100.1.4 remote-as 1  
  neighbor 10.100.1.4 update-source Loopback0  
  neighbor 10.100.1.4 route-reflector-client  
  neighbor 10.100.1.4 next-hop-self all  
  neighbor 10.100.1.4 send-label  
  
ip prefix-list prefix-list-ospf1-into-ospf2 seq 5 permit 10.100.1.1/32  
  
route-map ospf1-into-ospf2 permit 10  
  match ip address prefix-list prefix-list-ospf1-into-ospf2
```

## RR2

```
interface Loopback0  
  ip address 10.100.1.2 255.255.255.255  
  
router ospf 1  
  network 10.1.0.0 0.0.255.255 area 0  
  network 10.100.1.2 0.0.0.0 area 0  
!  
router ospf 3  
  redistribute ospf 1 subnets match internal route-map ospf1-into-ospf3  
  network 10.3.0.0 0.0.255.255 area 0  
!  
router bgp 1  
  bgp log-neighbor-changes  
  neighbor 10.100.1.1 remote-as 1  
  neighbor 10.100.1.1 update-source Loopback0  
  neighbor 10.100.1.1 next-hop-self all  
  neighbor 10.100.1.1 send-label  
  neighbor 10.100.1.5 remote-as 1  
  neighbor 10.100.1.5 update-source Loopback0  
  neighbor 10.100.1.5 route-reflector-client  
  neighbor 10.100.1.5 next-hop-self all  
  neighbor 10.100.1.5 send-label  
  
ip prefix-list prefix-list-ospf1-into-ospf3 seq 5 permit 10.100.1.2/32  
  
route-map ospf1-into-ospf3 permit 10  
  match ip address prefix-list prefix-list-ospf1-into-ospf3
```

## PE2

```
interface Loopback0  
  ip address 10.100.1.5 255.255.255.255  
  
interface Ethernet1/0  
  no ip address  
  xconnect 10.100.1.4 100 encapsulation mpls  
  
router ospf 3
```

```

network 10.3.0.0 0.0.255.255 area 0
network 10.100.1.5 0.0.0.0 area 0

router bgp 1
  bgp log-neighbor-changes
  network 10.100.1.5 mask 255.255.255.255
  neighbor 10.100.1.2 remote-as 1
  neighbor 10.100.1.2 update-source Loopback0
  neighbor 10.100.1.2 send-label

```

**附註：**使用路由對映將核心IGP(ospf 1)重新分配到聚合IGP(ospf2或ospf 3)。此路由對映允許RR的環回字首重新分配到聚合IGP。這是因為只將RR的回送首碼直接通告到核心IGP(ospf 1)。但是，在聚合IGP中也必須知道RR的環回字首，因此PE路由器上的BGP可以與RR的環回對等。

## 驗證

請參見圖2以驗證控制平面操作。

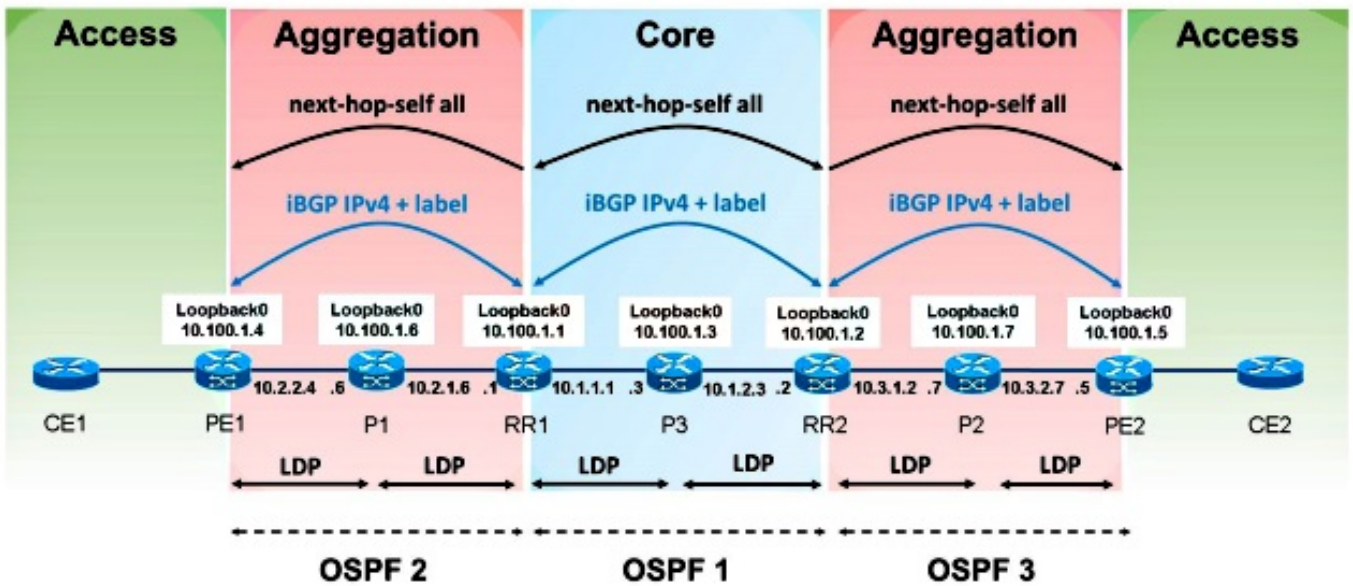


Figure 2

若要驗證MPLS標籤通告，請參閱圖3。

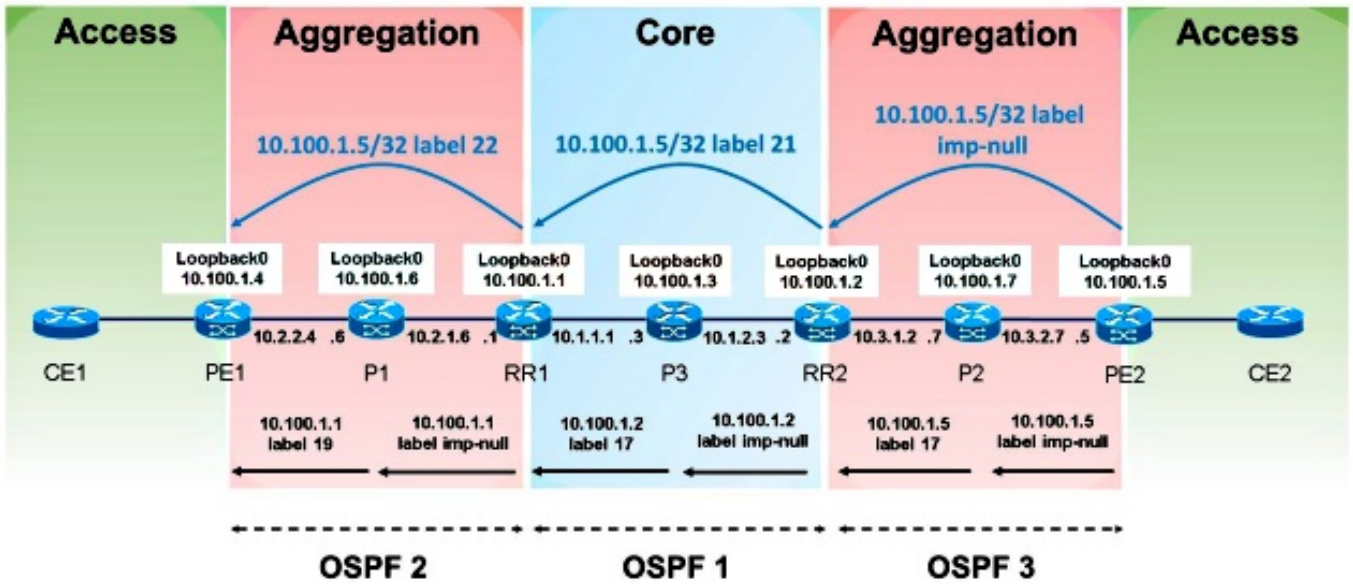


Figure 3

請參閱圖4以驗證封包轉送。

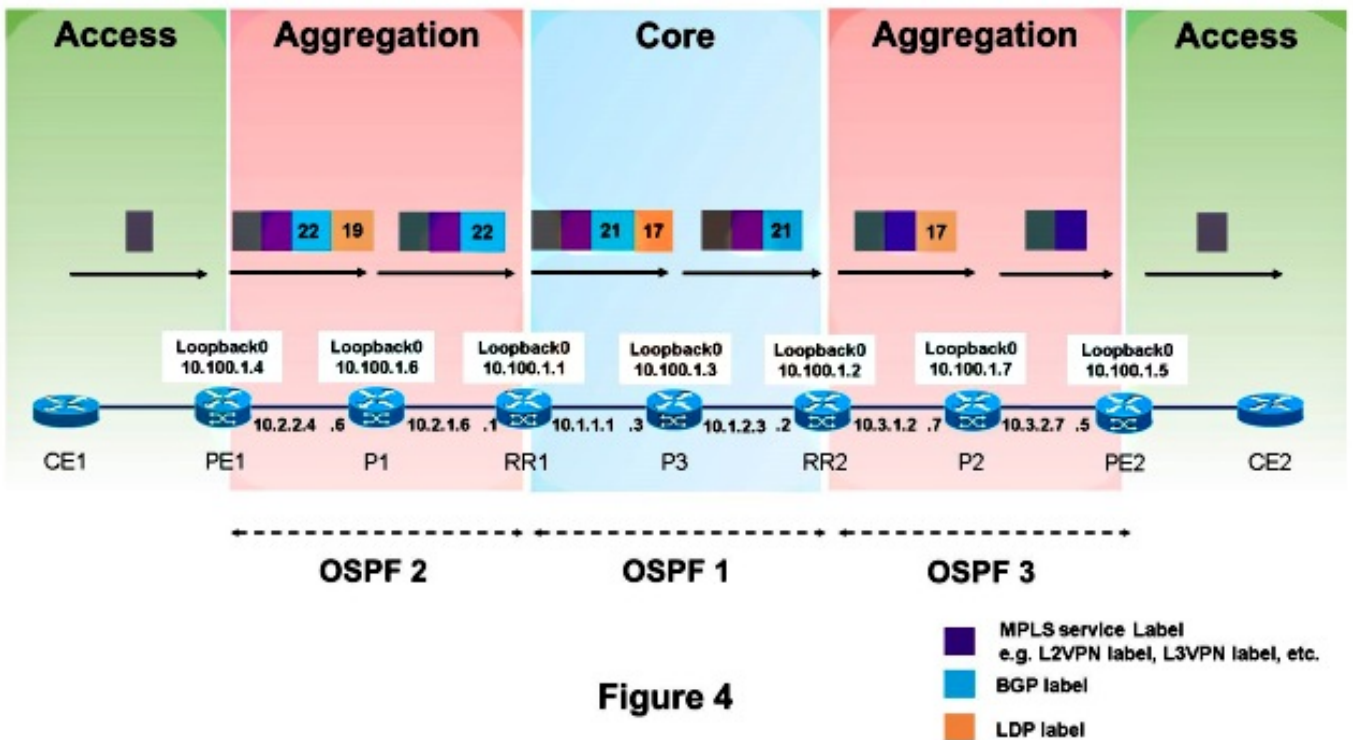


Figure 4

這是資料包從PE1轉發到PE2的方式。PE2的環回字首是10.100.1.5/32，因此該字首是重要的。

```
PE1#show ip route 10.100.1.5
```

```
Routing entry for 10.100.1.5/32
  Known via "bgp 1", distance 200, metric 0, type internal
  Last update from 10.100.1.1 00:11:12 ago
  Routing Descriptor Blocks:
    * 10.100.1.1, from 10.100.1.1, 00:11:12 ago
```

Route metric is 0, traffic share count is 1  
AS Hops 0  
**MPLS label: 22**

PE1#show ip cef 10.100.1.5  
10.100.1.5/32  
nexthop 10.2.2.6 Ethernet0/0 **label 19 22**

PE1#show ip cef 10.100.1.5 detail  
10.100.1.5/32, epoch 0, flags rib defined all labels  
1 RR source [no flags]  
**recursive via 10.100.1.1 label 22**  
nexthop 10.2.2.6 Ethernet0/0 **label 19**

PE1#show bgp ipv4 unicast labels

Network	Next Hop	In label/Out label
10.100.1.4/32	0.0.0.0	imp-null/nolabel
<b>10.100.1.5/32</b>	<b>10.100.1.1</b>	<b>nolabel/22</b>

P1#show mpls forwarding-table labels 19 detail

Local Label	Outgoing Label	Prefix or Tunnel Id	Bytes Label Switched	Outgoing interface	Next Hop
<b>19</b>	<b>Pop Label</b>	10.100.1.1/32	603468	Et1/0	10.2.1.1

MAC/Encaps=14/14, MRU=1504, Label Stack{  
AABBCC000101AABBCC0006018847  
No output feature configured

RR1#show mpls forwarding-table labels 22 detail

Local Label	Outgoing Label	Prefix or Tunnel Id	Bytes Label Switched	Outgoing interface	Next Hop
<b>22</b>	<b>21</b>	10.100.1.5/32	575278	Et0/0	10.1.1.3

MAC/Encaps=14/22, MRU=1496, **Label Stack{17 21}**  
AABBCC000300AABBCC0001008847 0001100000015000  
No output feature configured

RR1#show bgp ipv4 unicast labels

Network	Next Hop	In label/Out label
10.100.1.4/32	10.100.1.4	19/imp-null
<b>10.100.1.5/32</b>	<b>10.100.1.2</b>	<b>22/21</b>

P3#show mpls forwarding-table labels 17 detail

Local Label	Outgoing Label	Prefix or Tunnel Id	Bytes Label Switched	Outgoing interface	Next Hop
<b>17</b>	<b>Pop Label</b>	10.100.1.2/32	664306	Et1/0	10.1.2.2

MAC/Encaps=14/14, MRU=1504, Label Stack{  
AABBCC000201AABBCC0003018847  
No output feature configured

RR2#show mpls forwarding-table labels 21 detail

Local Label	Outgoing Label	Prefix or Tunnel Id	Bytes Label Switched	Outgoing interface	Next Hop
<b>21</b>	<b>17</b>	10.100.1.5/32	615958	Et0/0	10.3.1.7

MAC/Encaps=14/18, MRU=1500, **Label Stack{17}**  
AABBCC000700AABBCC0002008847 00011000  
No output feature configured

```
RR2#show bgp ipv4 unicast labels
```

```
Network          Next Hop          In label/Out label
10.100.1.4/32    10.100.1.1       22/19
10.100.1.5/32    10.100.1.5       21/imp-null
```

```
P2#show mpls forwarding-table labels 17 detail
```

```
Local      Outgoing  Prefix           Bytes Label  Outgoing  Next Hop
Label      Label     or Tunnel Id     Switched     interface
17         Pop Label  10.100.1.5/32    639957       Et1/0     10.3.2.5
MAC/Encaps=14/14, MRU=1504, Label Stack{}
AABBCC000500AABBCC0007018847
No output feature configured
```

```
PE1#trace
```

```
Protocol [ip]:
```

```
Target IP address: 10.100.1.5
```

```
Source address: 10.100.1.4
```

```
DSCP Value [0]:
```

```
Numeric display [n]:
```

```
Timeout in seconds [3]:
```

```
Probe count [3]:
```

```
Minimum Time to Live [1]:
```

```
Maximum Time to Live [30]:
```

```
Port Number [33434]:
```

```
Loose, Strict, Record, Timestamp, Verbose[none]:
```

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

```
Tracing the route to 10.100.1.5
```

```
VRF info: (vrf in name/id, vrf out name/id)
```

```
 1 10.2.2.6 [MPLS: Labels 19/22 Exp 0] 3 msec 3 msec 3 msec
 2 10.2.1.1 [MPLS: Label 22 Exp 0] 3 msec 3 msec 3 msec
 3 10.1.1.3 [MPLS: Labels 17/21 Exp 0] 3 msec 3 msec 2 msec
 4 10.1.2.2 [MPLS: Label 21 Exp 0] 2 msec 3 msec 2 msec
 5 * * *
 6 10.3.2.5 4 msec * 4 msec
```

附註：第5跳顯示？ 5 \* \* ?。這是因為路由器P2沒有用於traceroute的源IP地址 10.100.1.4(PE1)的路由。因此，路由器P2無法向PE1傳送網際網路控制消息協定(ICMP)錯誤消息。這是正常的，因為Unified MPLS的要點是，在一個聚合部分中沒有所有PE路由器的環回字首顯示在其他聚合部分的IGP中。路由器P2不會嘗試轉髮帶有原始標籤堆疊的ICMP錯誤消息。這是因為原始標籤堆疊只有一個標籤。如果封包的這個原始標籤堆疊具有兩個或多個標籤，則會沿LSP轉送ICMP錯誤訊息，且可能會傳回traceroute的來源。如果原始標籤堆疊只有一個標籤，則生成ICMP錯誤消息的路由器會嘗試路由查詢，並嘗試使用路由表（不使用原始標籤堆疊）對其進行路由。

```
P2#show ip route 10.100.1.4
```

```
% Subnet not in table
```

## 疑難排解

目前尚無適用於此組態的具體疑難排解資訊。

## 相關資訊



- [無縫MPLS架構](#)
- [技術支援與文件 - Cisco Systems](#)