

## HCIP-Datcom 分解实验 - MPLS

臧家林制作



MPLS 实验 1：MPLS 和 LDP 基本配置

MPLS 实验 2：BGP/MPLS VPN 基本配置

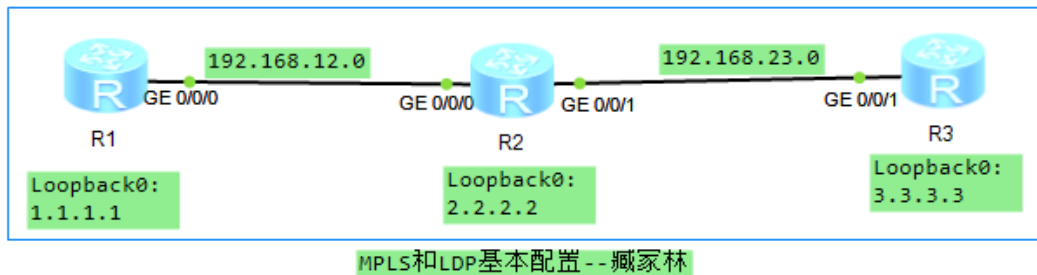
=====

### **MPLS 实验 1：MPLS 和 LDP 基本配置**

MPLS（Multi-Protocol Label Switching, 多协议标签交换）技术的出现，极大地推动了互联网的发展和应用。例如，利用 MPLS 技术，可以有效而灵活地部署 VPN（Virtual Private Network, 虚拟专用网）流量工程、TE（Traffic Engineering，流量工程）和 QoS（Quality of Service, 服务质量）。目前，MPLS 技术主要应用在运营商网络之中。

在 MPLS 网络中，位于网络边缘的路由器称为 LER（Label Edge Router），网络内部的路由称为 LSR（Label Switch Router），MPLS 报文经过的路由称为 LSP（Label Switched Path）。一条 LSP 总是起于一台被称为 Ingress 的 LER，止于另一台被称为 Egress 的 LER，中间经过若干台被称为 Transit 的 LSR。LSP 具有单向性，且有静态 LSP 和动态 LSP 之分，静态 LSP 需要人工进行固定的标签分配，动态 LSP 需

要利用诸如 LDP ( Label Distribution Protocol ,标签分发协议 ) 这样的协议进行动态标签分配。



### 基本配置 , OSPF 协议

R1:

```
undo ter mo
```

```
sys
```

```
sysname R1
```

```
user-interface console 0
```

```
idle-timeout 0 0
```

```
int loop 0
```

```
ip add 1.1.1.1 32
```

```
int g0/0/0
```

```
ip add 192.168.12.1 24
```

```
q
```

```
ospf router-id 1.1.1.1
```

```
area 0
```

```
net 192.168.12.1 0.0.0.0
```

```
net 1.1.1.1 0.0.0.0
```

```
q
```

R2:

```
undo ter mo
```

```
sys
```

```
sysname R2
user-interface console 0
idle-timeout 0 0
int loop 0
ip add 2.2.2.2 32
int g0/0/0
ip add 192.168.12.2 24
int g0/0/1
ip add 192.168.23.2 24
q
ospf router-id 2.2.2.2
area 0
net 192.168.12.2 0.0.0.0
net 192.168.23.2 0.0.0.0
net 2.2.2.2 0.0.0.0
q
```

```
R3:
undo ter mo
sys
sysname R3
user-interface console 0
idle-timeout 0 0
int loop 0
ip add 3.3.3.3 32
int g0/0/1
ip add 192.168.23.3 24
q
ospf router-id 3.3.3.3
area 0
```

```
net 192.168.23.3 0.0.0.0
net 3.3.3.3 0.0.0.0
q
```

配置完成后，查看 R1 的 OSPF 路由表，能看其他 2 台设备的环回口

```
[R1]dis ip routing-table
```

Destination/Mask	Proto	Pre
Cost	Flags	NextHop
Interface		
1.1.1.1/32	Direct	0
127.0.0.1	LoopBack0	0
2.2.2.2/32	OSPF	1
192.168.12.2		GigabitEthernet0/0/0
3.3.3.3/32	OSPF	2
192.168.12.2		GigabitEthernet0/0/0

=====

### 配置 MPLS 协议

配置 MPLS 协议，首先需要配置 LSR ID，全局开启 MPLS，在转发 MPLS 的接口上开启 MPLS 功能  
手动建立一条的静态 LSP

配置 LSR ID 用来在网络中唯一标识一个 MPLS 路由器。缺省没有配置 LSR ID，必须手工配置。为了提高网络的可靠性，推荐使用 LSR 某个 Loopback 接口的地址作为 LSR ID。

R1：

```
mpls lsr-id 1.1.1.1
mpls
int g0/0/0
mpls
q
static-lsp ingress R1toR3 destination 3.3.3.3 32 nexthop 192.
168.12.2 out-label 102
```

R2:

```
mpls lsr-id 2.2.2.2
mpls
int g0/0/0
mpls
int g0/0/1
mpls
q
static-lsp transit R1toR3 incoming-interface g0/0/0 in-label 1
02 nexthop 192.168.23.3 out-label 203
```

R3:

```
mpls lsr-id 3.3.3.3
mpls
int g0/0/1
mpls
q
static-lsp egress R1toR3 incoming-interface g0/0/1 in-label
203
```

配置完成后，在 R1 R2 R3 上查看一下，可以看到 R1 已经拥有了去往 R3 的静态 LSP，且在本地的 In 标签为 NULL，说

明 R1 是该 LSP 的 Ingress.

[R1]dis mpls lsp

-----  
-----

LSP

Information: STATIC LSP

-----  
-----

FEC In/Out

Label In/Out IF

Vrf Name

3.3.3.3/32

NULL/102

-/GE0/0/0

[R2]dis mpls lsp

-----  
-----

LSP

Information: STATIC LSP

-----  
-----

FEC In/Out

Label In/Out IF

Vrf Name

-/-

[R3]dis mpls lsp

-----  
-----

## LSP

Information: STATIC LSP

```
-----  
-----  
FEC                                         In/Out  
Label      In/Out IF  
Vrf Name  
-/-                                         203/NULL  
GE0/0/1/-
```

在 R1 上验证去往 R3 的 MPLS 报文所经过的路径

```
<R1>tracert lsp ip 3.3.3.3 32  
      LSP Trace Route FEC: IPV4 PREFIX  
3.3.3.3/32 , press CTRL_C to break.  
      TTL      Replier  
Time          Type          Downstream  
      0  
Ingress        192.168.12.2/[102 ]  
      1          192.168.12.2          30  
ms            Transit        192.168.23.3/[203 ]  
      2          3.3.3.3  
70 ms          Egress
```

在 R3 上验证去往 R1 的 MPLS 报文所经过的路径，可以看到系统提示 LSP 并不存在，说明 LSP 具有单向性。

```
<R3>tracert lsp ip 1.1.1.1 32  
Error: The specified LSP does not exist.  
<R3>
```

## 配置 R3 去往 R1 的静态 LSP

R3 :

```
static-lsp ingress R3toR1 destination 1.1.1.1 32 nexthop 192.168.23.2 out-label 302
```

R2:

```
static-lsp transit R3toR1 incoming-interface g0/0/1 in-label 302 nexthop 192.168.12.1 out-label 201
```

R1:

```
static-lsp egress R3toR1 incoming-interface g0/0/0 in-label 201
```

配置完成后，在 R3 上验证去往 R1 的 MPLS 报文所经过的路径

```
<R3>tracert lsp ip 1.1.1.1 32
```

```
<R3>tracert lsp ip 1.1.1.1 32
```

```
    LSP Trace Route FEC: IPV4 PREFIX  
1.1.1.1/32 , press CTRL_C to break.
```

TTL	Replier	
Time	Type	Downstream
0		
Ingress	192.168.23.2/[302]	
1	192.168.23.2	50
ms	Transit	192.168.12.1/[201]
2	1.1.1.1	
60 ms	Egress	



=====

## 利用 LDP 动态分发标签并建立 LSP

首先，在 R1 R2 R3 上删除之前创建的静态 LSP

R1:

```
undo static-lsp ingress R1toR3
```

```
undo static-lsp egress R3toR1
```

R2:

```
undo static-lsp transit R1toR3
```

```
undo static-lsp transit R3toR1
```

R3:

```
undo static-lsp egress R1toR3
```

```
undo static-lsp ingress R3toR1
```

使用 `mpls ldp` 命令全局启用 LDP，然后在接口上使用同样的命令在接口上开启 LDP

必须完成 MPLS 协议的配置，然后才能进行 LDP 的配置

R1 :

```
mpls ldp
```

```
int g0/0/0
```

```
mpls ldp
```

R2 :

```
mpls ldp
```

```
int g0/0/0
```

```
mpls ldp
```

```
int g0/0/1
```

```
mpls ldp
```

```
R3 :
mpls ldp
int g0/0/1
mpls ldp
q
```

在 R1 上查看 启用了 LDP 的接口 , R1 的 g0/0/0 启用了 LDP , 标签发布方式为 DU 方式  
[R1]dis mpls ldp int

LDP Interface Information in Public Network

Codes:LAM(Label Advertisement Mode),  
IFName(Interface name)

A '\*' before an interface means the entity is being deleted.

```
-----
-----
IFName                               Status
LAM      TransportAddress
HelloSent/Rcv

-----
-----
GE0/0/0                               Active
DU      1.1.1.1
12/10
-----
-----
```

查看 LDP 会话信息，Operational 表示会话已经成功建立。

LSR 在 LDP 会话中的角色：

Active：LSR ID 值较大的一方表示建立 LDP 会话的主动方。

Passive：LSR ID 值较小的一方表示建立 LDP 会话的被动方。

```
[R1]dis mpls ldp session
LDP Session(s) in Public Network
Codes: LAM(Label Advertisement Mode),
SsnAge Unit(DDDD:HH:MM)
A '*' before a session means the session
is being deleted.
```

```
-----
-----
PeerID                               Status
LAM   SsnRole   SsnAge
KASent/Rcv

-----
-----
2.2.2.2:0                            Operational DU
Passive      0000:00:01      7/7
-----
-----
```

```
[R2]dis mpls ldp session
```

```
-----
-----
PeerID                               Status
```

```
LAM      SsnRole      SsnAge
KASent/Rcv
```

```
-----
-----
1.1.1.1:0                                Operational
DU      Active      0000:00:02      10/10
3.3.3.3:0                                Operational
DU      Passive     0000:00:02      9/9
-----
-----
```

```
[R3]dis mpls ldp session
```

```
-----
-----
PeerID                                Status
LAM      SsnRole      SsnAge
KASent/Rcv
```

```
-----
-----
2.2.2.2:0                                Operational
DU      Active      0000:00:02      12/12
-----
-----
```

查看 LSP 信息，LDP 为 R1 去往 R3，R3 去往 R1 均动态建立了 LSP

```
[R1]dis mpls lsp
```

-----  
-----

LSP

Information: LDP LSP

-----  
-----

FEC Label	In/Out IF	In/Out
Vrf Name		
1.1.1.1/32		3/NULL
-/-		
2.2.2.2/32		NULL/3
-/GE0/0/0		
2.2.2.2/32		1024/3
-/GE0/0/0		
3.3.3.3/32		NULL/1025
-/GE0/0/0		
3.3.3.3/32		1025/1025
-/GE0/0/0		

R1 去往 R3 的为 NULL/1025 、 1025/3 、 3/NULL

R3 去往 R1 的为 NULL/1024 、 1024/3 、 3/NULL

在 R1 上验证去往 R3 的 MPLS 报文所经过的路径，R1 在出发时被赋予了标签 1025，经过 R2 时，标签被替换为 3

=====

标签管理的 3 种方式

[R1]dis mpls ldp interface

## LDP Interface Information in Public Network

Codes: LAM(Label Advertisement Mode),  
IFName(Interface name)

A '\*' before an interface means the entity is being deleted.

-----  
-----

IFName	Status
LAM	TransportAddress
HelloSent/Rcv	

-----  
-----

GE0/0/0	Active
DU	1.1.1.1
62/60	

-----  
-----

另一条查看命令  
display mpls ldp

-----  
-----

Instance ID	: 0
-------------	-----

```

VPN-Instance          :
  Instance Status      : Active
  LSR ID               :
  1.1.1.1
  Loop Detection       : Off
  Path Vector Limit    : 32
  Label Distribution Mode : Ordered
  Label Retention Mode : Liberal
  Instance Deleting State : No
  Instance Reseting State : No
  Graceful-Delete      : Off
  Graceful-Delete Timer : 5 Sec

```

```

-----
-----

```

可以查看到

标签发布方式 : DU ( Downstream Unsolicited , 下游自主方式 )

Label Advertisement Mode

标签的分配控制方式 : 有序标签分配控制方式      标签  
 的保持方式 : 自由标签保持方式

Label Distribution Mode : Ordered      Lab  
 el Retention Mode : Liberal

DU : 对于一个到达同一目地址报文的分组 , LSR 无需从上游获得标签请求消息即可进行标签分配与分发。

Ordered : 只有当该 LSR 已经具有此 IP 分组的下一跳的标签 , 或者该 LSR 就是该 IP 分组的出节点时 , 该 LSR 才可以向上

游发送此 IP 分组的标签。

Liberal：对于从邻居 LSR 收到的标签映射，无论邻居 LSR 是不是自己的下一跳都保留。

```
<R1>tracert lsp ip 3.3.3.3 32
      LSP Trace Route FEC: IPV4 PREFIX
3.3.3.3/32 , press CTRL_C to break.
      TTL      Replier
Time          Type          Downstream
      0
Ingress      192.168.12.2/[1025 ]
      1          192.168.12.2          70
ms          Transit      192.168.23.3/[3 ]
      2          3.3.3.3
30 ms      Egress
<R1>
```

R3 在出发时被赋予了标签 1024，经过 R2 时，标签被替换为 3

```
<R3>tracert lsp ip 1.1.1.1 32
      LSP Trace Route FEC: IPV4 PREFIX
1.1.1.1/32 , press CTRL_C to break.
      TTL      Replier
Time          Type          Downstream
      0
Ingress      192.168.23.2/[1024 ]
      1          192.168.23.2          30
ms          Transit      192.168.12.1/[3 ]
```



```
      2          1.1.1.1
50 ms      Egress
```

测试 MPLS 连通性，可以看到 R1 R3 之间可以通过 MPLS 的 LSP 进行报文的转发。

```
<R1>ping lsp ip 3.3.3.3 32
```

```
<R3>ping lsp ip 1.1.1.1 32
```

```
<R1>ping lsp ip 3.3.3.3 32
```

```
LSP PING FEC: IPV4 PREFIX 3.3.3.3/32/ : 100
data bytes, press CTRL_C
```

```
      Reply from 3.3.3.3: bytes=100
Sequence=1 time=60 ms
      Reply from 3.3.3.3: bytes=100
Sequence=2 time=60 ms
      Reply from 3.3.3.3: bytes=100
Sequence=3 time=40 ms
      Reply from 3.3.3.3: bytes=100
Sequence=4 time=100 ms
      Reply from 3.3.3.3: bytes=100
Sequence=5 time=50 ms
```

```
      --- FEC: IPV4 PREFIX 3.3.3.3/32 ping
statistics ---
```

```
      5 packet(s) transmitted
```

```
      5 packet(s) received
```

```
=====
```

**MPLS 实验 2 : BGP/MPLS VPN 基本配置**

BGP/MPLS VPN 有时也简称为 MPLS L3 VPN，它是 MPLS 最为广泛的应用之一。BGP/MPLS VPN 主要部署在运营商网络中。

在 BGP/MPLS VPN 网络中，路由器被分为 3 类：PE 路由器（Provide Edge Router）、P 路由器（Provide Router）和 CE 路由器（Customer Edge Router）。P 路由器为 BGP/MPLS VPN 网络内部的路由器，通过只需要运行 IGP、MPLS 和 LDP。PE 路由器为 BGP/MPLS VPN 网络的边缘路由器，用于连接客户的 CE 设备，通过需要运行 MP-BGP（Multi-Protocol BGP）、IGP、MPLS 和 LDP，并为不同的 VPN 客户配置 VPN 实例。CE 为客户的边缘设备，用于连接 PE，其上仅需要配置 PE-CE 连通性。

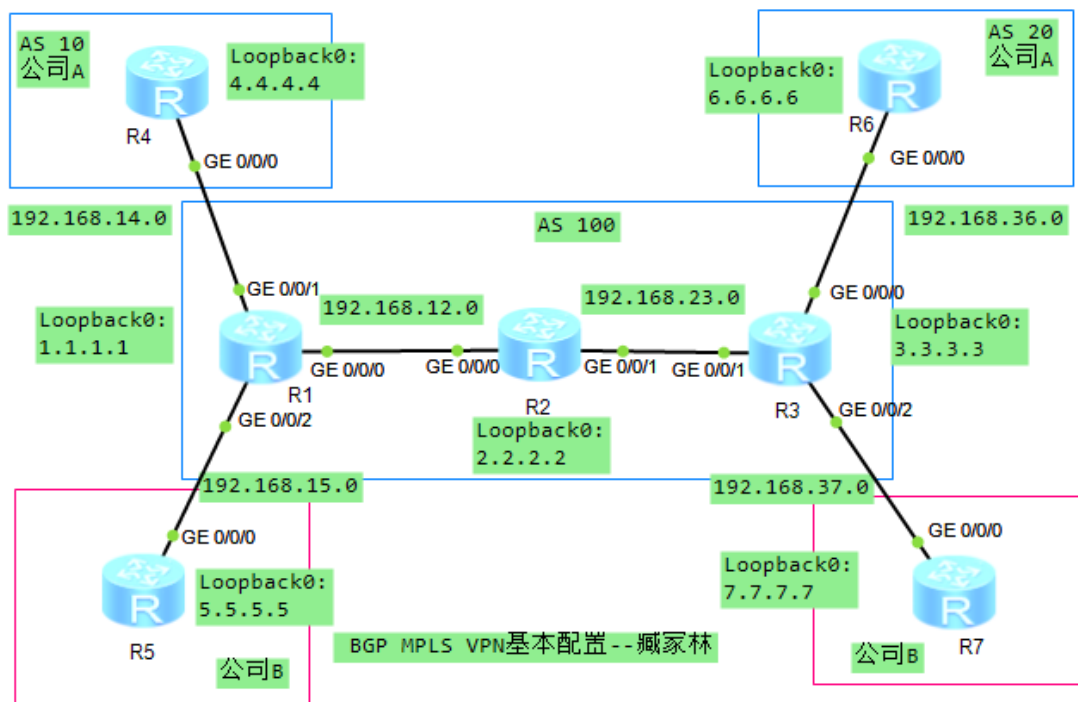
传统的 BGP 只能维护单一路由表的路由信息，无法为地址重叠的不同客户直接提供服务，所以在 BGP/MPLS VPN 中使用了 MP-BGP，它可以通过 VPNv4 地址族来区分不同客户的网络层地址信息，并使用 VPN 实例区分不同 VPN 客户的路由及流量。

在 BGP/MPLS VPN 中，每个 VPN 实例为相应的 VPN 客户单独维护了一张路由和转发表，称为 VRF（VPN Routing and Forwarding Table），不同的 VPN 实例间的路由是不能够互通的。在 PE 上，通过将连接 CE 的接口绑定至 VPN 实例，就可以区分不同 VPN 客户的路由。当 PE 将 VPN 路由传递至对端 PE 后，对端 PE 将使用 VPN 实例的 RD（Route Distinguisher）与 VPN Target 属性来区分 VPN 路由并将其分配至对应的 VPN 实例。

在 BGP/MPLS VPN 中，BGP 扩展团体属性 VPN Target 用来控制 VPN 路由的发布和接受。对于一个 VPN 实例，其 Export Target 与 Import Target 相互对应。一般情况下，对端 PE 上 VPN 实例的 Export Target 就与本地的 Import Target 相同，本地 VPN 实例的 Export Target 应与对端 PE 的 Import Target 相同。

通常，在 BGP/MPLS VPN 中，P 路由器无需运行 BGP，也无需知道关于 VPN 的任何信息。PE 上的 MP-BGP 会为 VPN 路由分配相应的标签值（VPN 标签），作为内层标签，LDP 分配的标签作为外层标签。当 VPN 流量沿 LSP 经过 P 路由器时，P 路由器只会进行外层标签的交换，当流量抵达对端 PE 时，对端 PE 会根据内层标签判断出流量所属的 VPN。

在 BGP/MPLS VPN 中，PE-CE 连通性的方式决定了客户如何使自己的路由进入 VPN 实例。通常，可以使用 BGP 在 CE 与 PE 间建立 EBGP 连接来实现 PE-CE 的连通，也可以使用静态路由方式或其他动态路由协议来实现这一目的。



## 基本 IP 配置

R1:

```
undo ter mo
```

```
sys
```

```
sysname R1
```

```
user-interface console 0
```

```
idle-timeout 0 0
```

```
int loop 0
```

```
ip add 1.1.1.1 32
```

```
int g0/0/1
```

```
ip add 192.168.14.1 24
```

```
int g0/0/2
```

```
ip add 192.168.12.1 24
```

```
int g0/0/2
```

```
ip add 192.168.15.1 24
```

```
q
```

```
R2:
undo ter mo
sys
sysname R2
user-interface console 0
idle-timeout 0 0
int loop 0
ip add 2.2.2.2 32
int g0/0/0
ip add 192.168.12.2 24
int g0/0/1
ip add 192.168.23.2 24
q
```

```
R3:
undo ter mo
sys
sysname R3
user-interface console 0
idle-timeout 0 0
int loop 0
ip add 3.3.3.3 32
int g0/0/0
ip add 192.168.36.3 24
int g0/0/1
ip add 192.168.23.3 24
int g0/0/2
ip add 192.168.37.3 24
q
```

```
R4:
undo ter mo
sys
sysname R4
```

```
user-interface console 0
idle-timeout 0 0
int loop 0
ip add 4.4.4.4 32
int g0/0/0
ip add 192.168.14.4 24
q
```

```
R5:
undo ter mo
sys
sysname R5
user-interface console 0
idle-timeout 0 0
int loop 0
ip add 5.5.5.5 32
int g0/0/0
ip add 192.168.15.5 24
q
```

```
R6:
undo ter mo
sys
sysname R6
user-interface console 0
idle-timeout 0 0
int loop 0
ip add 6.6.6.6 32
int g0/0/0
ip add 192.168.36.6 24
q
```

```
R7:
undo ter mo
```

```
sys
sysname R7
user-interface console 0
idle-timeout 0 0
int loop 0
ip add 7.7.7.7 32
int g0/0/0
ip add 192.168.37.7 24
q
```

### 配置运营商网络的 OSPF 协议

R1 R2 R3 在 AS 100 配置 OSPF 协议作为 IGP 协议

R1 :

```
ospf router-id 1.1.1.1
area 0
net 192.168.12.1 0.0.0.0
net 1.1.1.1 0.0.0.0
q
```

R2 :

```
ospf router-id 2.2.2.2
area 0
net 192.168.12.2 0.0.0.0
net 192.168.23.2 0.0.0.0
net 2.2.2.2 0.0.0.0
q
```

R3 :

```
ospf router-id 3.3.3.3
area 0
net 192.168.23.3 0.0.0.0
net 3.3.3.3 0.0.0.0
q
```

在 R2 上查看 OSPF 邻居的建立

```
[R2]dis ospf peer bri
```

OSPF Process 1 with Router ID 2.2.2.2  
Peer Statistic Information

-----	
-----	
Area Id	Interface
Neighbor id	State
0.0.0.0	
GigabitEthernet0/0/0	
1.1.1.1	Full
0.0.0.0	
GigabitEthernet0/0/1	
3.3.3.3	Full
-----	
-----	

配置 AS 100 网络的 MPLS 协议和 LDP

R1 R2 R3 配置 MPLS 和 LDP

```
R1 :  
mpls lsr-id 1.1.1.1  
mpls  
mpls ldp  
int g0/0/0  
mpls
```



```
mpls ldp
q
```

```
R2 :
mpls lsr-id 2.2.2.2
mpls
mpls ldp
int g0/0/0
mpls
mpls ldp
int g0/0/1
mpls
mpls ldp
q
```

```
R3 :
mpls lsr-id 3.3.3.3
mpls
mpls ldp
int g0/0/1
mpls
mpls ldp
q
```

配置完成后在 R2 上查看 LDP 会话建立情况，LDP 会话状态为 Operational，会话成功建立

```
[R2]dis mpls ldp session
```

LDP Session(s) in Public Network  
Codes: LAM(Label Advertisement Mode),  
SsnAge Unit(DDDD:HH:MM)  
A '\*' before a session means the session  
is being deleted.

-----						
-----						
PeerID			Status			
LAM	SsnRole	SsnAge				
KASent/Rcv						
-----						
-----						
1.1.1.1:0			Operational			
DU	Active	0000:00:00	3/3			
3.3.3.3:0			Operational			
DU	Passive	0000:00:00	2/2			
-----						
-----						

[R1]dis mpls lsp

```

-----
-----
                                     LSP
Information: LDP LSP
-----
-----
FEC                                     In/Out
Label      In/Out IF
Vrf Name
1.1.1.1/32                                3/NULL
-/-
2.2.2.2/32                                NULL/3
-/GE0/0/0

```

2.2.2.2/32	1024/3
-/GE0/0/0	
3.3.3.3/32	NULL/1025
-/GE0/0/0	
3.3.3.3/32	1025/1025
-/GE0/0/0	

### 配置 PE 设备间的 MP-BGP

在 R1 上建立与 R3 的 IBGP 邻居关系,使用 ipv4-family vpnv4 进入 VPNv4 视图,在 VPNv4 视图下启用与对等体交换 VPNv4 路由信息的能力,允许与对等体交换路由时携带 BGP 团体属性。

```
R1 :  
bgp 100  
router-id 1.1.1.1  
peer 3.3.3.3 as-number 100  
peer 3.3.3.3 connect-interface LoopBack 0  
peer 3.3.3.3 next-hop-local  
ipv4-family vpnv4  
peer 3.3.3.3 enable  
peer 3.3.3.3 advertise-community  
q
```

```
R3:  
bgp 100  
router-id 3.3.3.3  
peer 1.1.1.1 as-number 100  
peer 1.1.1.1 connect-interface LoopBack 0  
peer 1.1.1.1 next-hop-local  
ipv4-family vpnv4  
peer 1.1.1.1 enable
```

```
peer 1.1.1.1 advertise-community
q
```

配置完成后，在 R1 上查看 BGP 邻居关系，可以看到 R1 R3 邻居关系已经成功建立  
<R1>display bgp peer

Peer MsgSent	V OutQ	Up/Down	AS State	MsgRcvd PrefRcv
3.3.3.3	4	100		2
4	0	00:00:01	Established	
0				

### 在 PE 上创建 VPN 实例并与接口进行绑定

在 R1 上为公司 A 创建名为 vpna 的 VPN 实例，在 IPv4 地址族视图下使用 route-distinguisher 300:1 配置 RD 为 300:1，使用 vpn-target 100:1 both 配置 Import 与 Export 方向的 VPN-Target 团体属性。

将 g0/0/1 接口与 VPN 实例 vpna 进行绑定，绑定后接口的 IP 地址信息将被删除，需要重新配置。

```
R1 :
ip vpn-instance vpna
route-distinguisher 300:1
vpn-target 100:1 both
q

int g0/0/1
ip binding vpn-instance vpna
ip add 192.168.14.1 24
```

q

为公司 B 创建名为 vpnb 的 VPN 实例，配置 RD 为 300:2，vpn-target 为 100:2  
将 g0/0/2 接口与 VPN 实例 vpnb 进行绑定

R1 :

```
ip vpn-instance vpnb
route-distinguisher 300:2
vpn-target 100:2 both
q
```

```
int g0/0/2
ip binding vpn-instance vpnb
ip add 192.168.15.1 24
q
```

R3 上也完成相应的配置

R3 :

```
ip vpn-instance vpna
route-distinguisher 300:1
vpn-target 100:1 both
q
```

```
int g0/0/0
ip binding vpn-instance vpna
ip add 192.168.36.3 24
q
```

```
ip vpn-instance vpnb
route-distinguisher 300:2
vpn-target 100:2 both
q
```

```
int g0/0/2
ip binding vpn-instance vpnb
ip add 192.168.37.3 24
q
```

### 为公司 A 配置基于 BGP 的 PE-CE 连通性

在 CE R4 上进行 BGP 配置，建立与 R1 的 EBGP 邻居关系。

R4：

```
bgp 10
router-id 4.4.4.4
peer 192.168.14.1 as-number 100
network 4.4.4.4 32
```

在 R1 的 VPN 实例 vpna 的视图，建立与 R4 的 EBGP 邻居关系

R1：

```
bgp 100
ipv4-family vpn-instance vpna
peer 192.168.14.4 as-number 10
```

在 R4 上查看 BGP 邻居状态

```
[R4]dis bgp peer
```

Peer	V	AS	MsgRcvd
MsgSent	OutQ	Up/Down	State
Pre	fRcv		
192.168.14.1		4	100
2	2	1 00:00:00	
Established		0	

在 R1 上查看 VPN 实例 vpna 的 BGP 邻居状态

```
[R1]dis bgp vpnv4 vpn-instance vpna peer
```

```
BGP local router ID : 1.1.1.1
```

```
Local AS number : 100
```

```
VPN-Instance vpna, Router ID 1.1.1.1:
```

```
Total number of peers : 1                Peers  
in established state : 1
```

Peer	V	AS	MsgRcvd
MsgSent	OutQ	Up/Down	State Pre
fRcv			
192.168.14.4	4	10	3
2	0 00:00:03	Established	1

在 R3 和 R6 上完成同样的配置

R3 :

```
bgp 100
```

```
ipv4-family vpn-instance vpna
```

```
peer 192.168.36.6 as-number 20
```

R6 :

```
bgp 20
```

```
router-id 6.6.6.6
```

```
peer 192.168.36.3 as-number 100
```

```
net 6.6.6.6 32
```

q

在 R1 上查看 VPN 实例 vpna 的 BGP 路由表，可以看到 VPN 实例 vpna 仅拥有 4.4.4.4 6.6.6.6

```
<R1>dis bgp vpnv4 vpn-instance vpna
routing-table
```

VPN-Instance vpna, Router ID 1.1.1.1:

```
Total Number of Routes: 2
      Network
NextHop          MED
LocPrf          PrefVal Path/Ogn

* > 4.4.4.4/32
192.168.14.4      0
0                10i
* > i 6.6.6.6/32      3.3.3.3
0                100      0
20i
```

在 R1 上查看 LSP 信息

可以看到，表中出现了 BGP LSP, FEC 为 4.4.4.4/32，In 标签为 1027, Out 标签为 NULL, VRF Name 为 vpna，In 标签为 1027 应该由 MP-BGP 协议分配的内层标签，仅用于区分路由信息所属的 VRF

```
<R1>dis mpls lsp
```

```
-----
-----
```



LSP

Information: BGP LSP

-----  
-----

FEC In/Out

Label In/Out IF

Vrf Name

4.4.4.4/32 1027/NULL

-/-

vpna

-----  
-----

LSP

Information: LDP LSP

-----  
-----

FEC In/Out

Label In/Out IF

Vrf Name

1.1.1.1/32 3/NULL

-/-

2.2.2.2/32 NULL/3

-/GE0/0/0

2.2.2.2/32 1024/3

-/GE0/0/0

3.3.3.3/32 NULL/1025

-/GE0/0/0

3.3.3.3/32 1025/1025

-/GE0/0/0

在 R4 上 ping R6，测试连通性，可以看到 R4 与 R6 能正常通信，实现了公司 A 的 VPN 网络的互通。

```
<R4>ping -a 4.4.4.4 6.6.6.6
  PING 6.6.6.6: 56      data bytes, press
CTRL_C to break
    Reply from 6.6.6.6: bytes=56
Sequence=1 ttl=253 time=110 ms
    Reply from 6.6.6.6: bytes=56
Sequence=2 ttl=253 time=110 ms
    Reply from 6.6.6.6: bytes=56
Sequence=3 ttl=253 time=80 ms
    Reply from 6.6.6.6: bytes=56
Sequence=4 ttl=253 time=80 ms
    Reply from 6.6.6.6: bytes=56
Sequence=5 ttl=253 time=140 ms

    --- 6.6.6.6 ping statistics ---
        5 packet(s) transmitted
        5 packet(s) received
```

### 为公司 B 配置 PE-CE 连通性

公司 B 的 CE 设备 R5 将使用静态路由方式实现 PE-CE 连通性，CE 设备 R7 将使用 OSPF 协议实现 PE-CE 连通性

在 R5 上创建缺省路由

R5:

```
ip route-static 0.0.0.0 0 192.168.15.1
```

在 R1 上为 VPN 实例 vpnb 创建静态路由，将 VPN 实例 vpnb

的静态路由引入 BGP

至此，R1 与 R5 之间的 PE-CE 连通性配置完成

R1：

```
ip route-static vpn-instance vpnb 5.5.5.5 32 192.168.15.5
bgp 100
ipv4-family vpn-instance vpnb
import-route static
```

在 R7 上配置普通的 OSPF

R7:

```
ospf router-id 7.7.7.7
area 0
net 192.168.37.7 0.0.0.0
net 7.7.7.7 0.0.0.0
```

在 R3 上为 VPN 实例 vpnb 创建 OSPF 进程，要用 **ospf 2**

R3：

```
ospf 2 vpn-instance vpnb
area 0
net 192.168.37.3 0.0.0.0
q
```

配置完成后，在 R3 上查看 OSPF 邻居关系，状态为 FULL，  
表明邻居关系已经成功建立

<R3>display ospf peer brief

[R3]dis ospf peer bri

### OSPF Process 1 with Router ID 3.3.3.3

```
-----  
-----  
Area Id                               Interface  
Neighbor id                           State  
0.0.0.0  
GigabitEthernet0/0/1  
2.2.2.2                               Full
```

### OSPF Process 2 with Router ID 192.168.37.3

#### Peer Statistic Information

```
-----  
-----  
Area Id                               Interface  
Neighbor id                           State  
0.0.0.0  
GigabitEthernet0/0/2                 7.7.7.7  
Full
```

在 R3 的 OSPF 视图下使用 import-route bgp ，将 VPN 实例 vpnb 的 BGP 路由引入 OSPF  
然后，将 VPN 实例 vpnb 的 OSPF 路由引入 BGP,至此 R7 与 R3 之间的 PE-CE 连通性配置完成。

```
R3 :  
ospf 2  
import-route bgp  
q
```

```
bgp 100  
ipv4-family vpn-instance vpnb  
import-route ospf 2
```

如果只引入 7.7.7.7 接口，可以在 R3 上进行过滤

```
R3 :  
acl 2000  
rule permit source 7.7.7.7 0
```

```
route-policy 10 permit node 10  
if-match acl 2000
```

```
bgp 100  
ipv4-family vpn-instance vpnb  
import-route ospf 2 route-policy 10
```

在 R7 上测试与 R5 的连通性。R7 仅能与同属公司 B 的 5.5.5.5 进行通信，而不能与公司 A 通信。

```
<R7>ping -a 7.7.7.7 5.5.5.5  
    PING 5.5.5.5: 56      data bytes, press  
CTRL_C to break  
        Reply from 5.5.5.5: bytes=56  
Sequence=1 ttl=253 time=110 ms  
        Reply from 5.5.5.5: bytes=56  
Sequence=2 ttl=253 time=110 ms
```

```
Reply from 5.5.5.5: bytes=56
Sequence=3 ttl=253 time=110 ms
Reply from 5.5.5.5: bytes=56
Sequence=4 ttl=253 time=130 ms
Reply from 5.5.5.5: bytes=56
Sequence=5 ttl=253 time=110 ms
```

当 CE-PE 之间运行 EBGp 时，无需在 PE 上对客户路由和 MP-BGP 协议之间进行引入配置，客户的 VPNv4 路由可以直接通过 MPLS/MP-BGP 网络传递给对端 PE。

而当 CE-PE 之间运行的是静态路由或者是 IGP 时，则需要进行互相引入的配置，才能使客户的 VPNv4 路由通过 MPLS/MP-BGP 网络进行传递。