

MPLS VPN over ATM: with BGP or RIP on the Customer Site

Document ID: 10468

Contents

Introduction

Prerequisites

- Hardware and Software Versions

- Conventions

Background Information

- Description

Configure Procedure

- Network Diagram

- Configuration Procedure Part I

- Configuration Procedure Part II

Configurations

Show Commands

- Routing-Specific Commands

- MPLS Labels

- Address Overlapping

Sample Debug Output

Related Information

Introduction

This document provides a sample configuration of a Multiprotocol Label Switching (MPLS) VPN over ATM when Border Gateway Protocol (BGP) or Routing Information Protocol (RIP) is present on customer sites.

The Virtual Private Network (VPN) feature, when used with MPLS, allows several sites to transparently interconnect through a service provider network. One service provider network can support several different IP VPNs. Each of these appears to its users as a private network, separate from all other networks. Within a VPN, each site can send IP packets to any other site in the same VPN.

Each VPN is associated with one or more VPN routing or forwarding instances (VRFs). A VRF consists of an IP routing table, derived Cisco express forwarding (CEF) table, and set of interfaces that uses this forwarding table.

The router maintains a separate routing and CEF table for each VRF. This does not allow information to be sent outside the VPN, but it allows the same subnet to be used in several VPNs without duplicate IP address problems.

The router that uses BGP distributes the VPN routing information with the BGP extended communities.

For more information about the propagation of updates through a VPN, see these links:

- VPN Route Target Communities.
- BGP Distribution of VPN Routing Information.
- MPLS Forwarding.

Prerequisites

Hardware and Software Versions

These letters represent the different types of routers and switches used:

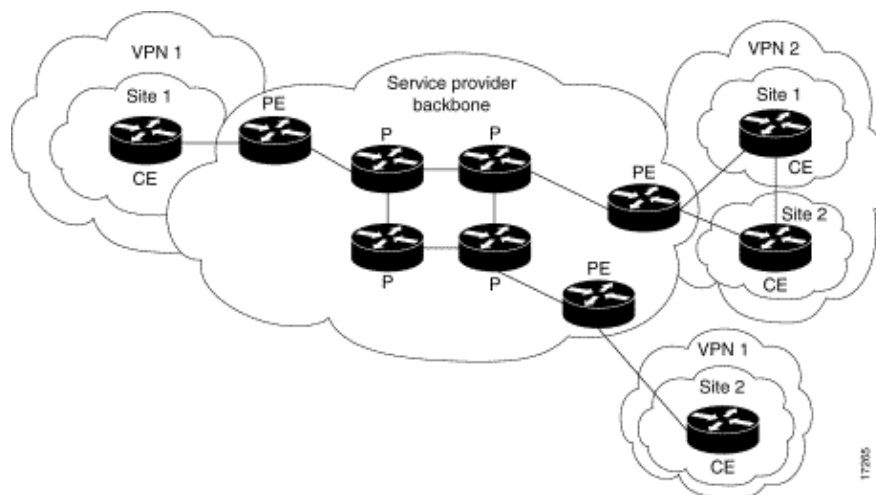
- **P** : Provider core router
- **PE** : Provider edge router
- **CE** : Customer edge router
- **C** : Customer router

We developed and tested the configuration with these software and hardware versions:

- **PE routers:**
 - ♦ **Software:** Cisco IOS® Software Release 12.1(3)T. Release 12.0(5)T includes the MPLS VPN.
 - ♦ **Hardware:** Any Cisco router from the 3600 series or higher, such as the Cisco 3660 or 7206.
- **CE routers:** Use any router that is able to exchange routing information with its PE router.
- **P routers and switches:** The MPLS VPN integration function resides only on the edge of the MPLS network, so use any MPLS-capable switch. In the sample configuration, the MPLS cloud is composed of an 8540 MSR and a LightStream 1010. If you use the LightStream 1010, we recommend that you use software version WA4.8d or higher. You can also use other ATM switches, such as the Cisco BPX 8650 or MGX 8850 in the ATM core network.

Conventions

This diagram shows a typical configuration that illustrates these conventions:



Refer to Cisco Technical Tips Conventions for more information on document conventions.

Background Information

Description

We have set up a standard MPLS ATM backbone with Open Shortest Path First (OSPF) area 0 as the Interior Gateway Protocol (IGP). We have configured two different VPNs with this backbone. The first of these uses

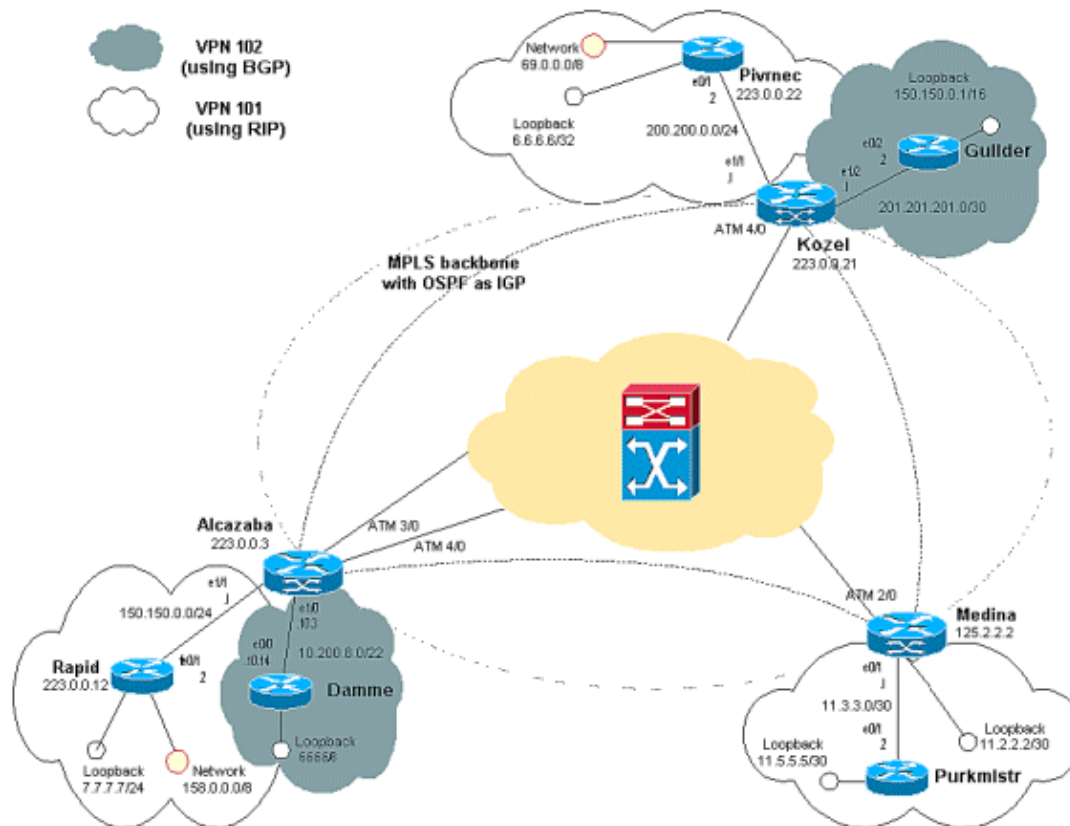
RIP as its customer–edge to provider–edge (CE–PE) routing protocol, and the other uses BGP as its PE–CE routing protocol.

We have configured various loopback and static routes on the CE routers to simulate the presence of other routers and networks.

Configure Procedure

Note: It is mandatory to use BGP as the VPN IGP between PE routers. This is because the use of BGP extended communities is the only way to transport routing information for the VPN between the PE routers.

Network Diagram



Configuration Procedure Part I

The Cisco IOS documentation (MPLS Virtual Private Networks) also describes this configuration procedure.

Make sure that ip cef is enabled. If you use a Cisco 7500 router, ensure that ip cef distributed is enabled. On the PEs, once MPLS has been set up, follow these steps:

1. Create one VRF for each VPN connected with the ip vrf <VPN routing/forwarding instance name> command:
 - a. Specify the correct route distinguisher used for that VPN. This is used to extend the IP address so that you can identify the VPN to which it belongs.

```
rd <VPN route distinguisher>
```
 - b. Set up the import and export properties for the BGP extended communities. These are used to filter the import and export process.

```
route-target [export|import|both] <target VPN extended community>
```

2. Configure the forwarding details for the respective interfaces with this command:

```
ip vrf forwarding <table name>
```

Note: Remember to set up the IP address after you do this.

3. Dependent on the PE–CE routing protocol that you use, you must now do one or more of these:

- ◆ Configure the static routes:

```
ip route vrf vrf-name prefix mask [next-hop-address] [interface {interface-name}]
```

- ◆ Configure the RIP with this command:

```
address-family ipv4 vrf <VPN routing/forwarding instance name>
```

Once you have done this part, enter the normal RIP configuration commands.

Note: This is only applied to the forwarding interfaces for the current VRF.

Note: You have to redistribute the correct BGP into RIP. When you do this, remember to also specify the metric that is used.

- ◆ Declare the BGP neighbor information.
- ◆ Configure the OSPF with the new IOS command:

```
router ospf <process ID> vrf <VPN routing/forwarding instance name>.
```

Note: This is only applied to the forwarding interfaces for the current VRF.

Note: You have to redistribute the correct BGP into OSPF. When you do this, remember to also specify the metric that is used.

Note: Once you attribute the OSPF process to a VRF, this process number is always used for this particular VRF. This applies even if you do not specify it in the command line.

Configuration Procedure Part II

Configure BGP between the PE routers. There are several ways to configure BGP; one way is to use the route reflector or confederation methods. The method used here—direct neighbor configuration—is the simplest and least scalable.

1. Declare the different neighbors.
2. Enter the address-family ipv4 vrf <VPN routing/forwarding instance name> for each VPN present at this PE router. Carry out one or more of these steps, as necessary:
 - ◆ Redistribute the static routing information.
 - ◆ Redistribute the RIP routing informations.
 - ◆ Redistribute the OSPF routing information.
 - ◆ Activate BGP in proximity to the CE routers.
3. Enter the address-family vpnv4 mode, and perform one of these:
 - ◆ Activate the neighbors.
 - ◆ Specify that extended community must be used. This is mandatory.

Configurations

In the Alcazaba configuration, lines specific to the VPN 101 are shown in bold, those specific to the VPN 102 are in italics, and those specific to both are shown in both bold and italics.

Alcazaba
<pre>! ip vrf vrf101 rd 1:101 route-target export 1:101 route-target import 1:101 ! ip vrf vrf102 rd 1:102 route-target export 1:102 route-target import 1:102 ! ip cef ! interface Loopback0 ip address 223.0.0.3 255.255.255.255 ! <i>interface Ethernet1/0</i> <i>ip vrf forwarding vrf102</i> <i>ip address 10.200.10.3 255.255.252.0</i> ! interface Ethernet1/1 ip vrf forwarding vrf101 ip address 150.150.0.1 255.255.255.0 ! interface ATM3/0 no ip address no ip mroute-cache no atm ilmi-keepalive pvc qsaal 0/5 qsaal pvc ilmi 0/16 ilmi ! ! interface ATM3/0.1 tag-switching ip address 10.0.0.17 255.255.255.252 tag-switching atm vpi 2-4 tag-switching ip ! interface ATM4/0 no ip address no atm ilmi-keepalive ! interface ATM4/0.1 tag-switching ip address 10.0.0.13 255.255.255.252 tag-switching atm vpi 2-4 tag-switching ip ! router ospf 1 network 10.0.0.0 0.0.0.255 area 0 network 223.0.0.3 0.0.0.0 area 0 ! router rip version 2 ! address-family ipv4 vrf vrf101 version 2 redistribute bgp 1 metric 0</pre>

```

network 150.150.0.0
no auto-summary
exit-address-family
!
router bgp 1
no synchronization
neighbor 125.2.2.2 remote-as 1
neighbor 125.2.2.2 update-source Loopback0

neighbor 223.0.0.21 remote-as 1

neighbor 223.0.0.21 update-source Loopback0

no auto-summary
!
address-family ipv4 vrf vrf102
redistribute connected
neighbor 10.200.10.14 remote-as 158
neighbor 10.200.10.14 activate
no auto-summary
no synchronization
exit-address-family
!
address-family ipv4 vrf vrf101
redistribute rip
no auto-summary
no synchronization
exit-address-family
!
address-family vpv4
neighbor 125.2.2.2 activate
neighbor 125.2.2.2 send-community extended

neighbor 223.0.0.21 activate
neighbor 223.0.0.21 send-community extended

no auto-summary
exit-address-family
!

```

Kozel

```

!
ip vrf vrf101
rd 1:101
route-target export 1:101
route-target import 1:101
!
ip vrf vrf102
rd 1:102
route-target export 1:102
route-target import 1:102
!
ip cef
!
interface Loopback0
ip address 223.0.0.21 255.255.255.255
!
interface Ethernet1/1
ip vrf forwarding vrf101
ip address 200.200.0.1 255.255.255.0
!

```

```

interface Ethernet1/2
 ip vrf forwarding vrf102
 ip address 201.201.201.1 255.255.255.252
!
interface ATM4/0
 no ip address
 no atm scrambling cell-payload
 no atm ilmi-keepalive
 pvc qsaal 0/5 qsaal
 pvc ilmi 0/16 ilmi
!
interface ATM4/0.1 tag-switching
 ip address 10.0.0.6 255.255.255.252
 tag-switching atm vpi 2-4
 tag-switching ip
!
router ospf 1
 log-adjacency-changes
 network 10.0.0.0 0.0.0.255 area 0
 network 223.0.0.21 0.0.0.0 area 0
!
router rip
 version 2
!
 address-family ipv4 vrf vrf101
 version 2
 redistribute bgp 1 metric 1
 network 200.200.0.0
 no auto-summary
 exit-address-family
!
router bgp 1
 no synchronization
 neighbor 125.2.2.2 remote-as 1
 neighbor 125.2.2.2 update-source Loopback0
 neighbor 223.0.0.3 remote-as 1
 neighbor 223.0.0.3 update-source Loopback0
 no auto-summary
!
 address-family ipv4 vrf vrf102
 redistribute connected
 redistribute static
 neighbor 201.201.201.2 remote-as 69
 neighbor 201.201.201.2 activate
 no auto-summary
 no synchronization
 exit-address-family
!
 address-family ipv4 vrf vrf101
 redistribute rip
 no auto-summary
 no synchronization
 exit-address-family
!
 address-family vpnv4
 neighbor 125.2.2.2 activate
 neighbor 125.2.2.2 send-community extended
 neighbor 223.0.0.3 activate
 neighbor 223.0.0.3 send-community extended
 no auto-summary
 exit-address-family
!

```

```
Current configuration:
!
ip vrf vrf101
  rd 1:101
  route-target export 1:101
  route-target import 1:101
ip cef
!
interface Loopback1
  ip vrf forwarding vrf101
  ip address 11.2.2.2 255.255.255.252
!
interface ATM2/0
  no ip address
  no atm ilmi-keepalive
!
interface ATM2/0.66 tag-switching
  ip address 125.1.4.2 255.255.255.252
  tag-switching ip
!
interface Ethernet1/1
  ip vrf forwarding vrf101
  ip address 11.3.3.1 255.255.255.252
!
router ospf 1

  network 125.1.4.0 0.0.0.3 area 0
  network 125.2.2.2 0.0.0.0 area 0
!
router rip
  version 2
  network 11.0.0.0
  !
  address-family ipv4 vrf vrf101
  version 2
  redistribute bgp 1 metric 1
  network 11.0.0.0
  no auto-summary
  exit-address-family
!
router bgp 1
  no synchronization
  neighbor 223.0.0.3 remote-as 1
  neighbor 223.0.0.3 update-source Loopback0
  neighbor 223.0.0.21 remote-as 1
  neighbor 223.0.0.21 update-source Loopback0
  !
  address-family ipv4 vrf vrf101
  redistribute connected
  redistribute static
  redistribute rip
  default-information originate
  no auto-summary
  no synchronization
  exit-address-family
  !
  address-family vpnv4
  neighbor 223.0.0.3 activate
  neighbor 223.0.0.3 send-community extended
  neighbor 223.0.0.21 activate
  neighbor 223.0.0.21 send-community extended
  exit-address-family
!
```


Rapid

Current configuration:

```
!  
interface Loopback0  
 ip address 223.0.0.12 255.255.255.255  
!  
interface Loopback2  
 ip address 7.7.7.7 255.255.255.0  
!  
interface FastEthernet0/1  
 ip address 150.150.0.2 255.255.255.0  
 duplex auto  
 speed auto  
!  
router rip  
 version 2  
 redistribute static  
 network 7.0.0.0  
 network 10.0.0.0  
 network 150.150.0.0  
 no auto-summary  
!  
ip route 158.0.0.0 255.0.0.0 Null  
!
```

Damme

```
!  
interface Loopback1  
 ip address 6.6.6.6 255.0.0.0  
!  
interface FastEthernet0/0  
 ip address 10.200.10.14 255.255.252.0  
 duplex auto  
 speed autoa  
!  
router bgp 158  
 no synchronization  
 network 6.0.0.0  
 network 10.200.0.0 mask 255.255.252.0  
 neighbor 10.200.10.3 remote-as 1  
 no auto-summary  
!
```

Pivrtec

Current configuration:

```
!  
interface Loopback0  
 ip address 223.0.0.22 255.255.255.255  
!  
interface Loopback1  
 ip address 6.6.6.6 255.255.255.255  
!  
interface FastEthernet0/1  
 ip address 200.200.0.2 255.255.255.0  
 duplex auto  
 speed auto  
!  
router rip  
 version 2  
 redistribute static
```

```
network 6.0.0.0
network 200.200.0.0
no auto-summary
!
ip route 69.0.0.0 255.0.0.0 Null0
!
```

Guilder

```
!
interface Loopback2
 ip address 150.150.0.1 255.255.0.0
!
interface Ethernet0/2
 ip address 201.201.201.2 255.255.255.252
!
router bgp 69
 no synchronization
 network 7.7.7.0 mask 255.255.0.0
 network 150.150.0.0
 network 201.201.201.0 mask 255.255.255.252
 redistribute connected
 neighbor 201.201.201.1 remote-as 1
 no auto-summary
!
```

Purkmister

```
Current configuration:
!
interface Loopback0
 ip address 11.5.5.5 255.255.255.252
!
interface FastEthernet0/1
 ip address 11.3.3.2 255.255.255.252
 duplex auto
 speed auto
!
router rip
 version 2
 network 11.0.0.0
!
```

Show Commands

Routing-Specific Commands

The Output Interpreter Tool (registered customers only) (OIT) supports certain **show** commands. Use the OIT to view an analysis of **show** command output.

- **show ip rip database vrf**
- **show ip bgp vpnv4 vrf**
- **show ip route vrf**
- **show ip route**

On a PE router, the PE-CE routing method (such as RIP, BGP or static) and the PE-PE BGP updates indicate the routing table that is used for a particular VRF. You can display the RIP information for a particular VRF:

```
Alcazaba#show ip rip database vrf vrf101
0.0.0.0/0 auto-summary
```

```

0.0.0.0/0
[2] via 150.150.0.2, 00:00:12, Ethernet1/1
6.0.0.0/8 auto-summary
6.6.6.6/32 redistributed
[1] via 223.0.0.21,
7.0.0.0/8 auto-summary
7.7.7.0/24
[1] via 150.150.0.2, 00:00:12, Ethernet1/1
10.0.0.0/8 auto-summary
10.0.0.0/8 redistributed
[1] via 125.2.2.2,
10.0.0.0/16
[1] via 150.150.0.2, 00:00:12, Ethernet1/1
10.200.8.0/22
[1] via 150.150.0.2, 00:00:12, Ethernet1/1
11.0.0.0/8 auto-summary
11.0.0.4/30 redistributed
[1] via 125.2.2.2,
11.1.1.0/30 redistributed
[1] via 125.2.2.2,
11.3.3.0/30 redistributed
[1] via 125.2.2.2,
11.5.5.4/30 redistributed
[1] via 125.2.2.2,
69.0.0.0/8 auto-summary
69.0.0.0/8 redistributed
[1] via 223.0.0.21,
150.150.0.0/16 auto-summary
150.150.0.0/24 directly connected, Ethernet1/1
158.0.0.0/8
[1] via 150.150.0.2, 00:00:17, Ethernet1/1
200.200.0.0/24 auto-summary
200.200.0.0/24 redistributed
[1] via 223.0.0.21,

```

You can also display the BGP information for a particular VRF with the **show ip bgp vpv4 vrf** command. The PE-PE results from the internal BGP (IBGP) are indicated by an **i**.

```

Alcazaba#show ip bgp vpv4 vrf vrf101
  BGP table version is 46, local router ID is 223.0.0.3
  Status codes: s suppressed, d damped, h history, * valid, best, i - internal
  Origin codes: i - IGP, e - EGP, ? - incomplete

```

```

Network Next Hop Metric LocPrf Weight Path
Route Distinguisher: 1:101 (default for vrf vrf101)
 *i6.6.6.6/32 223.0.0.21 1 100 0 ?
 * 7.7.7.0/24 150.150.0.2 1 32768 ?
 * 10.0.0.0/16 150.150.0.2 1 32768 ?
 * 10.200.8.0/22 150.150.0.2 1 32768 ?
 *i11.2.2.0/30 125.2.2.2 0 100 0 ?
 *i11.3.3.0/30 125.2.2.2 0 100 0 ?
 *i11.5.5.4/30 125.2.2.2 1 100 0 ?
 *i69.0.0.0 223.0.0.21 1 100 0 ?
 * 150.150.0.0/24 0.0.0.0 0 32768 ?
 * 158.0.0.0/8 150.150.0.2 1 32768 ?
 *i200.200.0.0 223.0.0.21 0 100 0 ?

```

```

Kozel#show ip bgp vpv4 vrf vrf102
  BGP table version is 48, local router ID is 223.0.0.21
  Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
  Origin codes: i - IGP, e - EGP, ? - incomplete

```

```

      Network          Next Hop          Metric LocPrf Weight Path
Route Distinguisher: 1:102 (default for vrf vrf102)
 * i6.0.0.0            223.0.0.3          0      100      0 158 i

```

```

*>i          223.0.0.3          0    100    0 158 i
*> 7.7.0.0/16    201.201.201.2      0          0 69 ?
* 10.200.8.0/22  201.201.201.2      0          0 69 ?
* i          223.0.0.3          0    100    0 ?
*>i          223.0.0.3          0    100    0 ?
*> 102.102.0.0/16 201.201.201.2      0          0 69 ?
*> 150.150.0.0    201.201.201.2      0          0 69 i
* 201.201.201.0/30 201.201.201.2      0          0 69 i
*>          0.0.0.0          0          32768 ?

```

You can check the global routing table for a VRF on both the PE and the CE routers. These match. For the PE router, you have to specify the VRF with the **show ip route vrf** command .

```

Alcazaba#show ip route vrf vrf101
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
P - periodic downloaded static route
Gateway of last resort is not set
B 69.0.0.0/8 [200/1] via 223.0.0.21, 00:11:03
B 200.200.0.0/24 [200/0] via 223.0.0.21, 00:11:03
  6.0.0.0/32 is subnetted, 1 subnets
B 6.6.6.6 [200/1] via 223.0.0.21, 00:11:03
  7.0.0.0/24 is subnetted, 1 subnets
R 7.7.7.0 [120/1] via 150.150.0.2, 00:00:05, Ethernet1/1
  10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
R 10.0.0.0/16 [120/1] via 150.150.0.2, 00:00:05, Ethernet1/1
R 10.200.8.0/22 [120/1] via 150.150.0.2, 00:00:05, Ethernet1/1
  11.0.0.0/30 is subnetted, 3 subnets
B 11.3.3.0 [200/0] via 125.2.2.2, 00:07:05
B 11.2.2.0 [200/0] via 125.2.2.2, 00:07:05
B 11.5.5.4 [200/1] via 125.2.2.2, 00:07:05
  150.150.0.0/24 is subnetted, 1 subnets
C 150.150.0.0 is directly connected, Ethernet1/1
R 158.0.0.0/8 [120/1] via 150.150.0.2, 00:00:06, Ethernet1/1

```

For Pivrtec, this is the standard routing table, so use the **show ip route** command:

```

Pivrtec#show ip route
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
P - periodic downloaded static route Gateway of last resort is not set S 69.0.0.0/8 is o
223.0.0.0/32 is subnetted, 1 subnets
C 223.0.0.22 is directly connected, Loopback0
C 200.200.0.0/24 is directly connected, FastEthernet0/1
  6.0.0.0/32 is subnetted, 1 subnets
C 6.6.6.6 is directly connected, Loopback1
  7.0.0.0/24 is subnetted, 1 subnets
R 7.7.7.0 [120/1] via 200.200.0.1, 00:00:23, FastEthernet0/1
  10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
R 10.0.0.0/16 [120/1] via 200.200.0.1, 00:00:23, FastEthernet0/1
R 10.200.8.0/22 [120/1] via 200.200.0.1, 00:00:24, FastEthernet0/1
  11.0.0.0/30 is subnetted, 3 subnets
R 11.3.3.0 [120/1] via 200.200.0.1, 00:00:24, FastEthernet0/1
R 11.2.2.0 [120/1] via 200.200.0.1, 00:00:25, FastEthernet0/1
R 11.5.5.4 [120/1] via 200.200.0.1, 00:00:25, FastEthernet0/1
  150.150.0.0/24 is subnetted, 1 subnets

```

```
R 150.150.0.0 [120/1] via 200.200.0.1, 00:00:25, FastEthernet0/1
R 158.0.0.0/8 [120/1] via 200.200.0.1, 00:00:25, FastEthernet0/1
```

MPLS Labels

Check the label stack used for any particular route:

```
Alcazaba#show tag-switching forwarding-table vrf vrf101 11.5.5.5 detail
Local Outgoing Prefix Bytes tag Outgoing Next Hop
tag tag or VC or Tunnel Id switched interface
None 2/91 11.5.5.4/30 0 AT4/0.1 point2point
MAC/Encaps=4/12, MTU=4466, Tag Stack{2/91(vcd=69) 37}
00458847 0004500000025000
```

You can also use the normal commands to view the tag allocations and VPI/VCI relations here.

Address Overlapping

The same address can be used in different VPNs without interference with the others. In this example, the 6.6.6.6 address is connected twice, to Pivrnec in the VPN 101 and to Damme in the VPN 102. We can check this with **ping** on one site and **debug ip icmp** on the other site.

```
Guilder#ping 6.6.6.6
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 6.6.6.6, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/4/4 ms
```

```
Damme#debug ip icmp
ICMP packet debugging is on
6d22h: ICMP: echo reply sent, src 6.6.6.6, dst 201.201.201.2
6d22h: ICMP: echo reply sent, src 6.6.6.6, dst 201.201.201.2
6d22h: ICMP: echo reply sent, src 6.6.6.6, dst 201.201.201.2
6d22h: ICMP: echo reply sent, src 6.6.6.6, dst 201.201.201.2
6d22h: ICMP: echo reply sent, src 6.6.6.6, dst 201.201.201.2
```

Sample Debug Output

Sample output that uses the same configuration is available here.

Related Information

- [More MPLS over ATM Information](#)
- [More ATM Information](#)
- [Technical Support & Documentation – Cisco Systems](#)

[Contacts & Feedback](#) | [Help](#) | [Site Map](#)

© 2014 – 2015 Cisco Systems, Inc. All rights reserved. [Terms & Conditions](#) | [Privacy Statement](#) | [Cookie Policy](#) | [Trademarks of Cisco Systems, Inc.](#)