Configuring MPLS Basic VPN with RIP on Customer Side

Document ID: 13732

Introduction

This sample configuration shows a Multiprotocol Label Switching (MPLS) Virtual Private Network (VPN) when Routing Information Protocol (RIP) is present on the customer's side.

The VPN feature, when used with MPLS, allows several sites to transparently interconnect through a service provider's network. One service provider network can support several different IP VPNs. Each IP VPN appears as a private network, separate from all other networks. Each site in a VPN sends IP packets to other sites 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, a derived Cisco express forwarding (CEF) table, and a set of interfaces that use the forwarding table.

The router maintains a separate routing and CEF table for each VRF. This prevents information from being sent outside the VPN and allows the same subnet to be used in several VPNs without causing duplicate IP address problems.

The router using Border Gateway Protocol (BGP) distributes the VPN routing information using the BGP extended communities.

For more information regarding the propagation of updates through a VPN see the VPN Route Target Communities, BGP Distribution of VPN Routing Information, and MPLS Forwarding sections in MPLS Virtual Private Networks.

Prerequisites

Requirements

There are no specific prerequisites for this document.

Components Used

We developed and tested this configuration using the software and hardware versions below:

- **PE routers**: The MPLS VPN functionality resides in the PE routers. Use Feature Navigator II (registered customers only) to determine which hardware and software combinations you can use.
- **CE routers**: Use any router able to exchange routing information with its PE router.
- **P routers and switches**: In this document, ATM switches such as the MSR, the BPX and the MGX were used. However, because the document focuses on the MPLS VPN feature we could also have used frame based MPLS in the core with routers, such as the Cisco 12000.

The information presented in this document was created from devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If you are working in a live network, ensure that you understand the potential impact of any command before using it.

Network Description

We set up a standard MPLS ATM backbone using Open Shortest Path First (OSPF) area 0 as the Interior Gateway Protocol (IGP). We configured two different VPNs using this backbone. The first VPN uses RIP as its customer−edge to provider−edge (CE−PE) routing protocol; the other VPN uses BGP as its PE−CE routing protocol. We configured various loopback and static routes on the CE routers to simulate the presence of other routers and networks.

Note: BGP must be used as the VPN IGP between PE routers, since using BGP extended communities is the only way to transport routing information for the VPN between the PE routers.

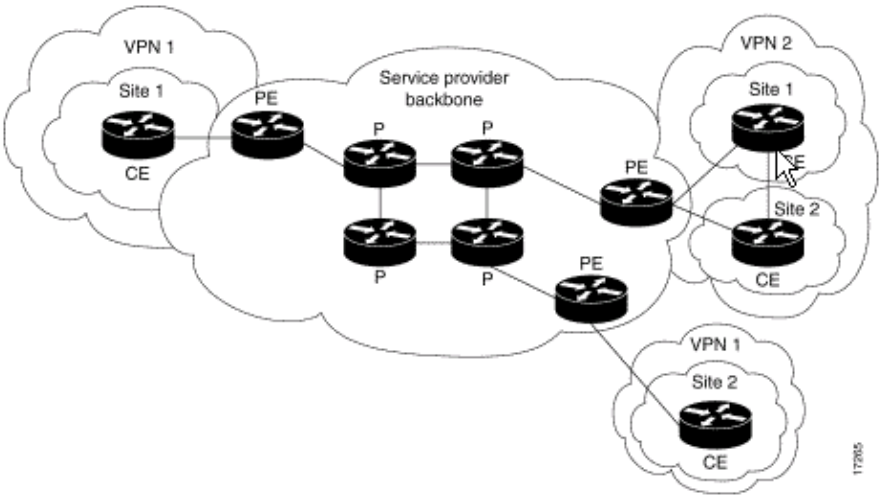
Note: An ATM network was used as the backbone network to make this configuration. This configuration applies to ATM (and other) protocol(s). PE routers must be able to reach each other using the MPLS network for the VPN configuration to work.

Conventions

The letters below represent the different types of routers and switches used:

- P: Provider's core router
- PE: Provider's edge router
- CE: Customer's edge router
- C: Customer's router

A typical configuration illustrating these conventions is shown in the diagram below:



For more information on document conventions, refer to Cisco Technical Tips Conventions.

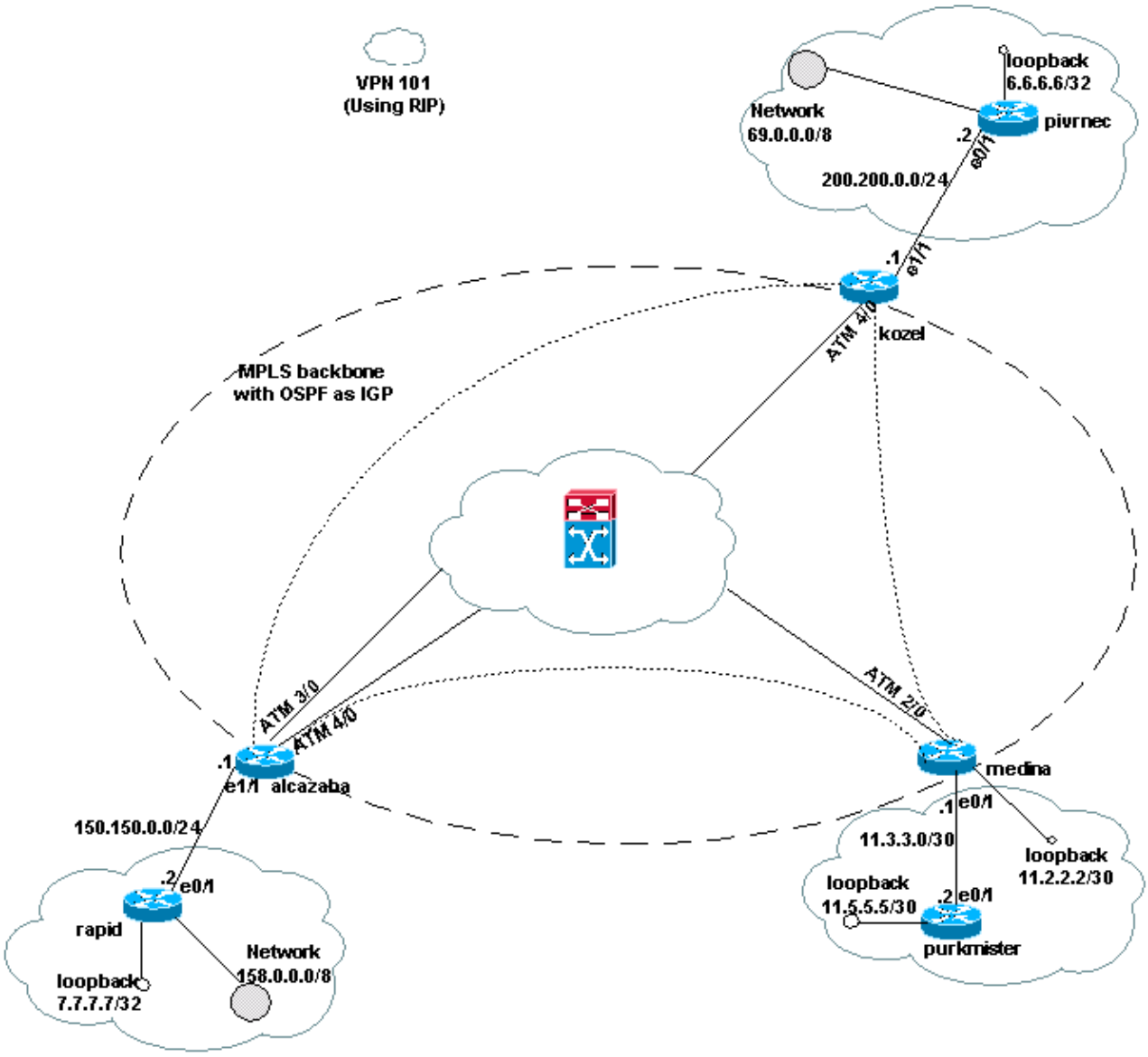
Configuration Procedure

In this section, you are presented with the information to configure the features described in this document. The Cisco IOS documentation found in MPLS Virtual Private Networks also describes this configuration procedure.

Note: To find additional information on the commands used in this document, use the IOS Command Lookup tool (registered customers only)

Network Diagram

This document uses the network setup shown in the diagram below.



Part I

The steps below will help you configure correctly.

Enable the **ip cef** command. If using a Cisco 7500 router, ensure that the **ip cef distributed** command is enabled, where available, to enhance performances on the PE, once MPLS is set up.

1. Create a VRF for each VPN using the **ip vrf [***VPN routing | forwarding instance name***]** command. While creating the VRFs, be sure to:

    - Specify the correct route distinguisher used for that VPN using the command below. The distinguisher is used to extend the IP address and allows you to identify to which VPN it belongs.

        rd [*VPN route distinguisher***]**
    - Set up the import and export properties for the BGP extended communities using the command below. These properties are used for filtering the import and export process.

        route-target {export | import | both} [*target VPN extended community***]**

2. Configure the forwarding details for the respective interfaces using the **ip vrf forwarding [***table name***]**command and remember to set up the IP address afterwards.
3. Depending on the PE−CE routing protocol used, do one or more of the following:

    - Configure the static routes as follows:

      ip route vrf *vrf−name prefix mask* **[***next−hop−address***] [***interface* **{***interface−nu*
    - Configure the RIP using the following command:

      address−family ipv4 vrf [*VPN routing* **|** *forwarding instance name***]**

      Once you have completed one or both of the steps above, enter the normal RIP configuration commands.

      Note: These commands apply only to the forwarding interfaces of the current VRF. Redistribute the correct BGP into RIP and remember to specify the metric used.
    - Declare the BGP neighbor information.
    - Configure the OSPF using the new IOS command:

      router ospf *process−id* **vrf [***VPN routing* | *forwarding instance name***]**

    Note: This command applies only to the forwarding interfaces for the current VRF. Redistribute the correct BGP routing information into OSPF and specify the metric used. Once the OSPF process to a VRF is complete, even if the OSPF process is not specified in the command line, this process ID is always used for this particular VRF.

Part II

Configure BGP between the PE routers. There are several ways to configure BGP, such as using the route reflector or confederation methods. The method shown here is direct neighbor configuration. It is the simplest and the least scalable.

1. Declare the different neighbors.
2. Enter the **address−family ipv4 vrf [***VPN routing* | *forwarding instance name***]** command for each VPN present at this PE router. Carry out one or more of the following steps, as necessary:

    - Redistribute the static routing information.
    - Redistribute the RIP routing information.
    - Redistribute the OSPF routing information.
    - Activate BGP neighboring with the CE routers.
3. Enter the **address−family vpnv4** mode and:

    - Activate the neighbors.
    - Specify that extended community must be used. This is mandatory.

Configuration Examples

In the Alcalzaba configuration, lines specific to the VPN configuration are shown in **bold**.

Alcazaba
!
ip vrf vrf101
rd 1:101
route−target export 1:101
route−target import 1:101

```
!
ip cef
!
interface Loopback0
 ip address 223.0.0.3 255.255.255.255
!
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
 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 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 223.0.0.21 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 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 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 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
!
```

Medina

```
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 Null0
!
```

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
!
```

Pivrnec

```
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
!
```

debug and show Commands

Before you use **debug** commands, refer to Important Information on Debug Commands. Routing−specific commands are listed here:

- **show ip rip database vrf** − Shows information contained in the RIP database for a particular VRF.
- **show ip bgp vpnv4 vrf** − Displays VPN address information from the BGP table.
- **show ip route vrf** − Displays the IP routing table associated with a VRF.
- **show ip route** − Displays all static IP routes, or those installed using the authentication, authorization, and accounting (AAA) route download function.

Certain **show** commands are supported by the Output Interpreter tool (registered customers only) , which allows you to view an analysis of **show** command output.

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 used for a particular VRF. You can display the RIP information for a particular VRF as follows:

```
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 display the BGP information for a particular VRF using the **show ip bgp vpnv4 vrf** command. The PE−PE results from the internal BGP (iBGP) are indicated by an i in the output below.

```
Alcazaba# show ip bgp vpnv4 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 ?
```

Check the global routing table for a VRF on both the PE and the CE routers. These VRFs should match. For the PE router, you have to specify the VRF using 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 − ISIS, L1 − ISIS level−1, L2 − ISIS level−2, IA − ISIS 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
```

The equivalent command on Pivrnec is the **show ip route** command, since for every customer (and customer edge) router this is the standard routing table.

```
Pivrnec# 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 − ISIS, L1 − ISIS level−1, L2 − ISIS level−2, IA − ISIS 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 directly connected, Null0
  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 route as follows:

```
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 use the normal commands for viewing the tag allocations along with the virtual path identifier and virtual channel identifier (VPI/VCI) relations as shown in How to Troubleshoot the MPLS VPN .

Address Overlapping

You can use the same address in different VPNs without interfering with other VPNs. 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 using the **ping** command on one site and the **debug ip icmp** command 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

Refer to Packet Flow in an MPLS VPN Environment to see sample output using the same configuration.

Troubleshoot

There is currently no specific troubleshooting information available for this configuration.

Related Information

- **Technical Support − Tools & Resources**
- **MPLS Support Page**
- **Technical Support − Cisco Systems**

Updated: Nov 16, 2007 Document ID: 13732