**Multi-Protocol Label Switching (MPLS)**

BINGNAN XU

ccnp (cisco certified network professional)

may. 2019

NEWPORT HIGH SCHOOL, WA

## Purpose

In this lab, I set up a **Multi-Protocol Label Switching (MPLS)** network with 3 Cisco Catalyst 3750 Metro Switches as MPLS core and use 2 routers on each side to imitate two local networks.

1. Communication between Local Routers and Switch Routers (provider edge, or **PE**) should use **EBGP (External Border Gateway Protocol)** as routing protocol.
2. Three Switch Router cores uses **OSPF** for internal communication.
3. The core should appear as a coherent service provide, which means **iBGP (Internal Border Gateway Protocol)** should be configured on all PE Switch routers.

Finally, the LDP packets between core routers will be captured and analyzed with Wireshark. Packets with configured MPLS labels should be found in the packet captures.

## Backgroud:

Many Compares MPLS to an VPN network, which makes MPLS confusing for the people trying to learn about them. The basic mechanism of MPLS is to add an MPLS label over the IP packet. Thus, the IP packets can be fast switched in a metropolitan switch network.

One of the advantages of MPLS is its efficiency in processing traffic. In traditional IP traffic, data is wrapped in IP headers—much like your cargo wrapped in a box. On the box, there are information like next-hop address, time-to-live label, etc. When a router receives such a box, it will read the information, unwrap it, and put the data into a new box, a new IP header—with the new next-hop address, new time-to-live label, etc.

However, MPLS uses tags to label incoming traffic. So, instead of the traditional “opening the packet, seeing what’s inside and wrapping it into a new packet” style routing, MPLS put tags on the outside of the packet. This label outside of IP header (layer 3) makes this packet a Layer 2 frame. Thus, the global ip address scheme is reduced into a much smaller network of tens, or maybe hundreds, of metropolitan switches.

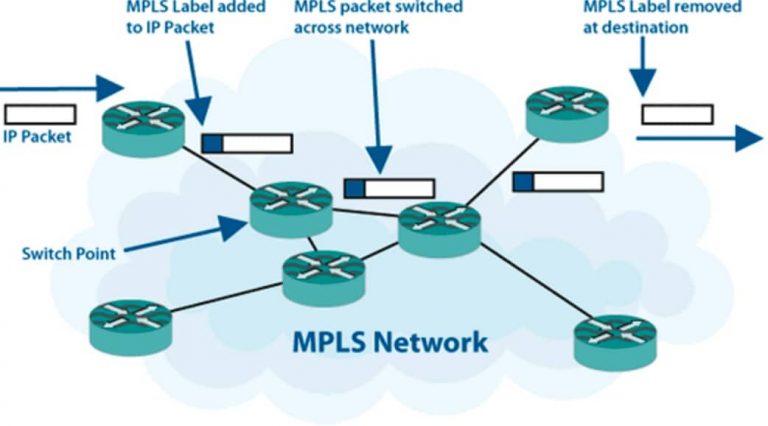


Image Credit: https://www.mushroomnetworks.com/mpls/

With MPLS, core networks can route traffic with much better efficiency. And since there is no need for new IP headers, traffic routing (or switching, in this sense, MPLS is a layer 2.5 protocol) in MPLS is much faster. Thus, MPLS is perfect for time sensitive service such as video conferencing and VOIP. To enhance the functionality in VOIP, MPLS has experimental bits, which are Quality of Service indicator that determine that priority of traffic (for example, voice is priority over video data, and video has priority over email data).

While MPLS is a Virtual Private Network, itself doesn’t provide encryption. MPLS is an IP packet wrapped in a switch label that only you can recognize. Normal VPN with extra headers for encryption consumes much more time than normal traffic, which yield them slow and unreliable for VOIP or video service. On the other hand, MPLS VPN is much faster, and more reliable, yet it’s costlier (require a lot of investment on infrastructure).

## Lab summary

First, I configure External Border Gateway Protocol **(EBGP)** on both sides of the Provide Edge (PE). After ensuring the connectivity on both sides (separately, between each PE and its local routers), I configure OSPF between the three core switch routers. At this point, the three core switch routers also established connectivity between each other.

Then, I redistribute OSPF route into BGP process and BGP routes into OSPF process, so that the connectivity of the whole network is established.

MPLS often operate with Provider Edges that runs BGP; thus, I will try this conventional practice, too. There are two types of BGP, EBGP and IBGP. The difference between EBGP and IBGP (Internal BGP) is that

* EBGP uses different AS numbers and must be directly connected. EBGP uses autonomous system path to prevent routing loops. It advertises routes like OSPF & EIGRP do.



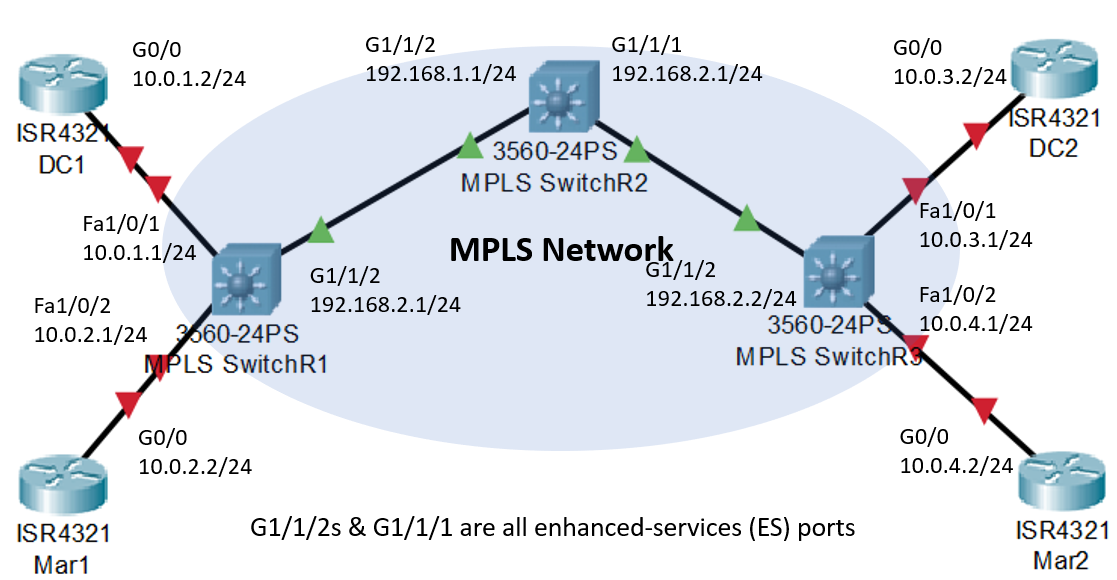
* IBGP uses the same Autonomous System numbers (AS) between neighbors. IBGP neighbors requires Full-Mesh Requirement required between each other. Because they don’t have distinct AS number to prevent routing loop

Globally unique AS numbers (1 - 64511) are assigned by InterNIC. Private autonomous system (AS) numbers which range from 64512 to 65535, that’s why my AS number is 650xx.

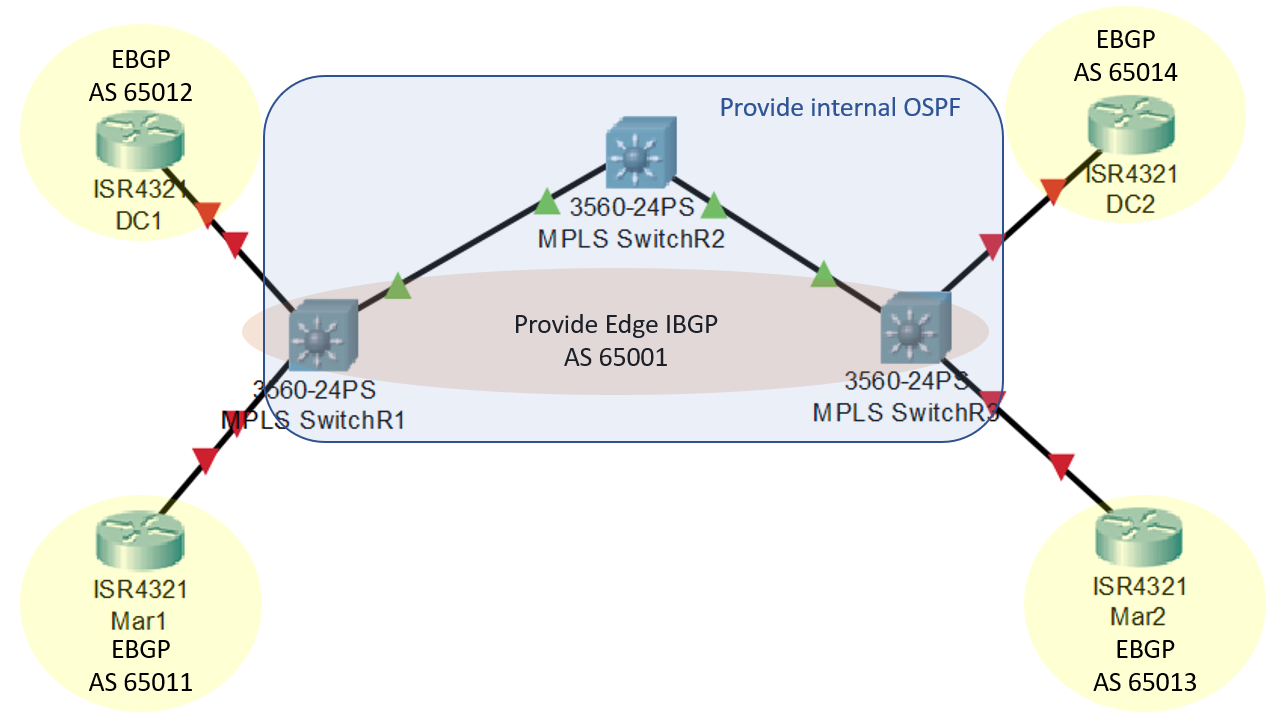
I use iBGP between two PE, because a real cooperation doesn’t get a lot of Globally unique AS number. I decide to only use one AS number across my core network.

## Lab Diagram

### IP Address scheme:



### Logical scheme:



## Lab commands

|  |  |
| --- | --- |
| MPLS Commands: | Function: |
| mpls label protocol ldp | MPLS uses LDP label (default) after this command |
| ip cef distributed | enables Cisco Express Forwarding, which is needed in MPLS switching |
| mpls ip | enable MPLS on this switch |
| mpls ldp advertise-labels | tells the switch to advertise LDP labels to other MPLS switches |
| mpls label range 20 40 | default MPLS labels starts from 16. I define a new label range (20 – 40) to distinguish my configuration in MPLS packet captures.  In real life, this command prevents conflicts with other MPLS systems that have overlapping label numbers. |
| mpls ldp router-id Loopback0 | Statically define MPLS router-id as the IP of Lo0. By default, MPLS use highest logical address, then it looks for highest operating address as its router-id. |
| interface GigabitEthernet1/1/2 | Enter the interface |
| mpls label protocol ldp | On this interface, MPLS uses LDP labels. |
| mpls ip | Start MPLS operation on this interface |

|  |  |
| --- | --- |
| BGP Commands: | Function: |
| eBGP: |  |
| router bgp 65001 | Create BGP Autonomous system 65001 |
| network 10.0.1.0 mask 255.255.255.0 | Define the adjacent network. like EIGRP |
| neighbor 10.0.1.2 remote-as 65011 | Neighbor should be the directly connected Interface of the neighbor. “remote-as” refers to the AS number of neighbors. |
| redistribute ospf 1 | Necessary for incorporating local OSPF network into BGP |
| neighbor 10.0.1.2 default-orig | Also necessary for distributing routes |
| router ospf 1 | Redistributing BGP routes into OSPF |
| redistribute bgp 65001 subnets | Redistribute BGP routes by its AS number |

|  |  |
| --- | --- |
|  |  |
| iBGP: |  |
| router bgp 65001 | Enter BGP Autonomous system 65001, the same one in EBGP |
| neighbor 192.168.3.3 remote-as 65001 | Form an IBGP relationship SwitchR3, which has a lo0 of 192.168.3.3 which is its BGP router ID |
| neighbor 192.168.3.3 update-source lo0 | Define that this SwitchR1 BGP router ID as it’s lo0 address, which is 192.168.3.1. |

## configuration

### Core MPLS switchrouters configuration:

**Switch(config)#** **SwitchR1**

ip routing

mpls label protocol ldp

ip cef distributed

mpls ip

mpls ldp advertise-labels

mpls label range 20 40

mpls ldp router-id Loopback0

interface Loopback0

ip address 192.168.3.1 255.255.255.255

interface GigabitEthernet1/1/2

no switchport

ip address 192.168.1.2 255.255.255.0

mpls label protocol ldp

mpls ip

int fa1/0/1

no switchport

ip address 10.0.1.1 255.255.255.0

mpls ip

int fa1/0/2

no switchport

ip address 10.0.2.1 255.255.255.0

mpls ip

router ospf 1

network 192.168.1.0 0.0.0.255 area 0

network 192.168.3.1 0.0.0.0 area 0

redistribute bgp 65001 subnets

router bgp 65001

network 10.0.1.0 mask 255.255.255.0

network 10.0.2.0 mask 255.255.255.0

neighbor 10.0.1.2 remote-as 65011

neighbor 10.0.2.2 remote-as 65012

neighbor 10.0.1.2 default-orig

neighbor 10.0.2.2 default-orig

neighbor 192.168.3.3 remote-as 65001

neighbor 192.168.3.3 update-source lo0

redistribute ospf 1

monitor session 1 source int G1/1/2

monitor session 1 source int fa1/0/1

monitor session 1 destination int fa1/0/24

**Switch(config)#** **SwitchR2Center**

ip routing

ip cef distributed

mpls ip

mpls label protocol ldp

mpls ldp advertise-labels

mpls ldp router-id Loopback0

mpls label range 20 40

interface Loopback0

ip address 192.168.3.2 255.255.255.255

interface GigabitEthernet1/1/1

no switchport

ip address 192.168.2.1 255.255.255.0

mpls label protocol ldp

mpls ip

interface GigabitEthernet1/1/2

no switchport

ip address 192.168.1.1 255.255.255.0

mpls label protocol ldp

mpls ip

router ospf 1

network 192.168.1.0 0.0.0.255 area 0

network 192.168.2.0 0.0.0.255 area 0

network 192.168.3.2 0.0.0.0 area 0

monitor session 1 source int g1/1/1- 2 both

monitor session 1 destination int fa1/0/24

**Switch(config)#** **hostname SwitchR3**

system mtu routing 1500

ip routing

ip cef distributed

mpls ip

mpls label protocol ldp

mpls ldp advertise-labels

mpls label range 20 40

mpls ldp router-id Loopback0

interface Loopback0

ip address 192.168.3.3 255.255.255.255

interface GigabitEthernet1/1/2

no switchport

ip address 192.168.2.2 255.255.255.0

mpls label protocol ldp

mpls ip

int fa1/0/1

no switchport

ip address 10.0.3.1 255.255.255.0

no shut

int fa1/0/2

no switchport

ip address 10.0.4.1 255.255.255.0

no shut

router ospf 1

network 192.168.2.0 0.0.0.255 area 0

network 192.168.3.3 0.0.0.0 area 0

redistribute bgp 65001 subnets

router bgp 65001

network 10.0.3.0

network 10.0.4.0

network 10.0.3.0 mask 255.255.255.0

network 10.0.4.0 mask 255.255.255.0

neighbor 10.0.3.2 remote-as 65013

neighbor 10.0.4.2 remote-as 65014

neighbor 10.0.3.2 default-orig

neighbor 10.0.4.2 default-orig

neighbor 192.168.3.1 remote-as 65001

neighbor 192.168.3.1 update-source lo0

redistribute ospf 1

### local ebpg routers configuration:

**Router(config)#** **hostname Mar1**

interface Loopback0

ip address 10.0.51.1 255.255.255.255

interface GigabitEthernet0/0

ip address 10.0.1.2 255.255.255.0

no shut

router bgp 65011

bgp log-neighbor-changes

network 10.0.1.0 mask 255.255.255.0

network 10.0.51.1 mask 255.255.255.255

**Router(config)#** **hostname DC1**

interface Loopback0

ip address 10.0.51.2 255.255.255.255

interface GigabitEthernet0/0

ip address 10.0.2.2 255.255.255.0

no shut

router bgp 65012

bgp log-neighbor-changes

network 10.0.2.0 mask 255.255.255.0

network 10.0.51.2 mask 255.255.255.255

neighbor 10.0.2.1 remote-as 65001

**Router(config)#** **hostname Mar2**

interface Loopback0

ip address 10.0.51.3 255.255.255.255

interface GigabitEthernet0/0

ip address 10.0.3.2 255.255.255.0

no shut

router bgp 65013

bgp log-neighbor-changes

network 10.0.3.0 mask 255.255.255.0

network 10.0.51.3 mask 255.255.255.255

neighbor 10.0.3.1 remote-as 65001

**Router(config)#** **hostname DC2**

interface Loopback0

ip address 10.0.51.4 255.255.255.255

interface GigabitEthernet0/0

ip address 10.0.4.2 255.255.255.0

no shut

router bgp 65014

bgp log-neighbor-changes

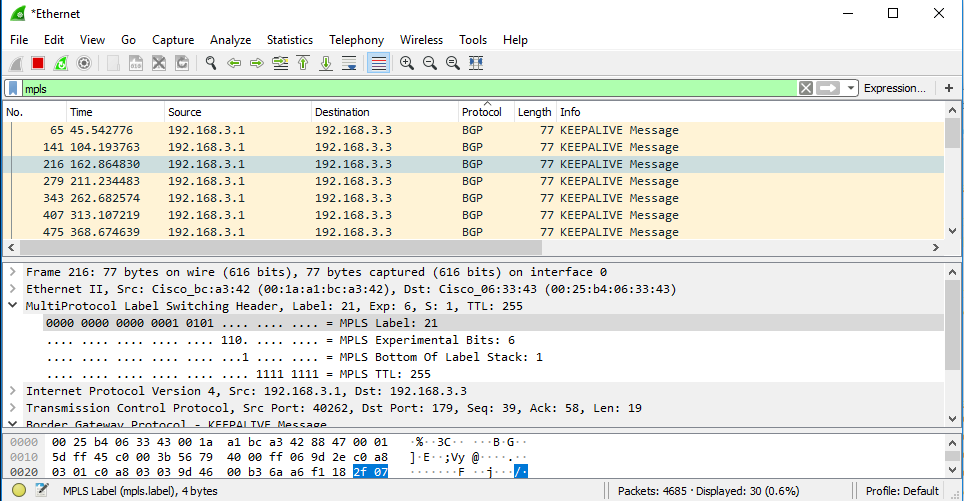
network 10.0.4.0 mask 255.255.255.0

network 10.0.51.4 mask 255.255.255.255

neighbor 10.0.4.1 remote-as 65001

## Wireshark captures:

On Port Fa1/0/24 of SwitchR1, we can find the BGP packets that it is sending to SwitchR3. You can see that the MPLS Label is 21, which is the first in my MPLS range 20-40. This packet is from SwitchR1 (192.168.3.1), and its destination is SwitchR3 (192.168.3.3). Such observation is also collaborated with the MPLS forwarding-table on SwitchR1:



**SwitchR1#show mpls forwarding-table**

Local Outgoing Prefix Bytes Label Outgoing Next Hop

Label Label or Tunnel Id Switched interface

20 Pop Label 192.168.3.2/32 0 Gi1/1/2 192.168.1.1

**21 Pop Label 192.168.2.0/24 0 Gi1/1/2 192.168.1.1**

**22 21 192.168.3.3/32 0 Gi1/1/2 192.168.1.1**

23 No Label 10.0.51.1/32 0 Fa1/0/1 10.0.1.2

24 No Label 10.0.51.2/32 0 Fa1/0/2 10.0.2.2

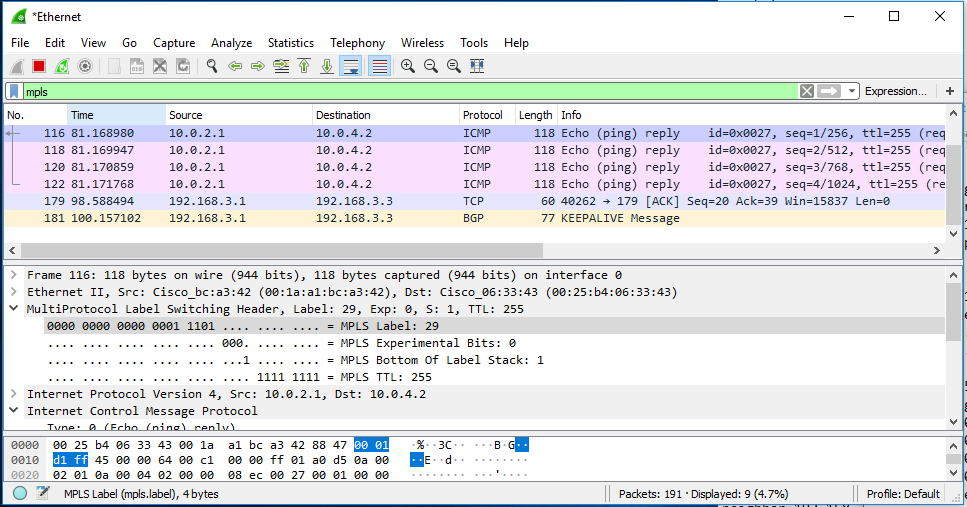
25 26 10.0.51.4/32 0 Gi1/1/2 192.168.1.1

26 27 10.0.51.3/32 0 Gi1/1/2 192.168.1.1

27 28 10.0.3.0/24 0 Gi1/1/2 192.168.1.1

28 29 10.0.4.0/24 0 Gi1/1/2 192.168.1.1

The outgoing label is 21. Its source is SwitchR1 (because this is a MPLS forwarding-table from SwitchR1, and the traffic comes from a local router of SwitchR1), and its destination is SwitchR3 (192.168.3.3).



**SwitchR2Center#show mpls forwarding-table**

Local Outgoing Prefix Bytes Label Outgoing Next Hop

Label Label or Tunnel Id Switched interface

20 Pop Label 192.168.3.1/32 13562 Gi1/1/2 192.168.1.2

21 Pop Label 192.168.3.3/32 15124 Gi1/1/1 192.168.2.2

22 24 10.0.51.2/32 0 Gi1/1/2 192.168.1.2

23 23 10.0.51.1/32 0 Gi1/1/2 192.168.1.2

24 Pop Label 10.0.1.0/24 3776 Gi1/1/2 192.168.1.2

25 Pop Label 10.0.2.0/24 4130 Gi1/1/2 192.168.1.2

26 28 10.0.51.4/32 0 Gi1/1/1 192.168.2.2

27 27 10.0.51.3/32 0 Gi1/1/1 192.168.2.2

28 Pop Label 10.0.3.0/24 0 Gi1/1/1 192.168.2.2

**29 Pop Label 10.0.4.0/24 5900 Gi1/1/1 192.168.2.2**

This is a ping (ICMP) packet capture done on Center SwitchR2. The source is from SwitchR1’s local network 10.0.2.0/24. The destination is to other side SiwthcR3’s local network 10.0.4.0/24. (a little bit confusing: **G1/1/1 is the port connected SwitchR2**, SwitchR3’s G1/1/2 address is 192.168.2.2)

Conclusion: the information from both sides with label 29 collaborate with each other. MPLS labels are working as expected.

## other show commands:

### ip routes for local routers:

**Mar1#show ip route**

Gateway of last resort is 10.0.1.1 to network 0.0.0.0

**\*the default gateway is set to the port directly connected to PE**

B\* 0.0.0.0/0 [20/0] via 10.0.1.1, 01:09:00

10.0.0.0/8 is variably subnetted, 9 subnets, 2 masks

C 10.0.1.0/24 is directly connected, GigabitEthernet0/0

L 10.0.1.2/32 is directly connected, GigabitEthernet0/0

B 10.0.2.0/24 [20/0] via 10.0.1.1, 00:54:48

B 10.0.3.0/24 [20/0] via 10.0.1.1, 00:12:10

B 10.0.4.0/24 [20/0] via 10.0.1.1, 00:11:39

C 10.0.51.1/32 is directly connected, Loopback0

B 10.0.51.2/32 [20/0] via 10.0.1.1, 01:09:00

B 10.0.51.3/32 [20/0] via 10.0.1.1, 00:11:39

B 10.0.51.4/32 [20/0] via 10.0.1.1, 00:11:39

B 192.168.1.0/24 [20/0] via 10.0.1.1, 00:54:48

B 192.168.2.0/24 [20/2] via 10.0.1.1, 00:54:48

192.168.3.0/32 is subnetted, 3 subnets

B 192.168.3.1 [20/0] via 10.0.1.1, 00:54:48

B 192.168.3.2 [20/2] via 10.0.1.1, 00:54:48

B 192.168.3.3 [20/3] via 10.0.1.1, 00:54:48

**DC1#show ip route**

**Gateway of last resort is 10.0.2.1 to network 0.0.0.0**

B\* 0.0.0.0/0 [20/0] via 10.0.2.1, 01:08:35

10.0.0.0/8 is variably subnetted, 9 subnets, 2 masks

B 10.0.1.0/24 [20/0] via 10.0.2.1, 00:54:54

C 10.0.2.0/24 is directly connected, GigabitEthernet0/0

L 10.0.2.2/32 is directly connected, GigabitEthernet0/0

B 10.0.3.0/24 [20/0] via 10.0.2.1, 00:11:44

B 10.0.4.0/24 [20/0] via 10.0.2.1, 00:11:14

B 10.0.51.1/32 [20/0] via 10.0.2.1, 01:08:35

C 10.0.51.2/32 is directly connected, Loopback0

B 10.0.51.3/32 [20/0] via 10.0.2.1, 00:11:14

B 10.0.51.4/32 [20/0] via 10.0.2.1, 00:11:14

B 192.168.1.0/24 [20/0] via 10.0.2.1, 00:54:23

B 192.168.2.0/24 [20/2] via 10.0.2.1, 00:54:23

192.168.3.0/32 is subnetted, 3 subnets

B 192.168.3.1 [20/0] via 10.0.2.1, 00:54:23

B 192.168.3.2 [20/2] via 10.0.2.1, 00:54:23

B 192.168.3.3 [20/3] via 10.0.2.1, 00:54:23

**Mar2#show ip route**

**Gateway of last resort is 10.0.3.1 to network 0.0.0.0**

B\* 0.0.0.0/0 [20/0] via 10.0.3.1, 00:30:29

10.0.0.0/8 is variably subnetted, 9 subnets, 2 masks

B 10.0.1.0/24 [20/0] via 10.0.3.1, 00:30:29

B 10.0.2.0/24 [20/0] via 10.0.3.1, 00:30:29

C 10.0.3.0/24 is directly connected, GigabitEthernet0/0

L 10.0.3.2/32 is directly connected, GigabitEthernet0/0

B 10.0.4.0/24 [20/0] via 10.0.3.1, 00:10:43

B 10.0.51.1/32 [20/0] via 10.0.3.1, 00:30:29

B 10.0.51.2/32 [20/0] via 10.0.3.1, 00:30:29

C 10.0.51.3/32 is directly connected, Loopback0

B 10.0.51.4/32 [20/0] via 10.0.3.1, 00:29:59

B 192.168.1.0/24 [20/2] via 10.0.3.1, 00:30:29

B 192.168.2.0/24 [20/0] via 10.0.3.1, 00:30:29

192.168.3.0/32 is subnetted, 3 subnets

B 192.168.3.1 [20/3] via 10.0.3.1, 00:30:29

B 192.168.3.2 [20/2] via 10.0.3.1, 00:30:29

B 192.168.3.3 [20/0] via 10.0.3.1, 00:30:29

**DC2#show ip route**

**Gateway of last resort is 10.0.4.1 to network 0.0.0.0**

B\* 0.0.0.0/0 [20/0] via 10.0.4.1, 00:29:55

10.0.0.0/8 is variably subnetted, 9 subnets, 2 masks

B 10.0.1.0/24 [20/0] via 10.0.4.1, 00:29:55

B 10.0.2.0/24 [20/0] via 10.0.4.1, 00:29:55

B 10.0.3.0/24 [20/0] via 10.0.4.1, 00:10:46

C 10.0.4.0/24 is directly connected, GigabitEthernet0/0

L 10.0.4.2/32 is directly connected, GigabitEthernet0/0

B 10.0.51.1/32 [20/0] via 10.0.4.1, 00:29:55

B 10.0.51.2/32 [20/0] via 10.0.4.1, 00:29:55

B 10.0.51.3/32 [20/0] via 10.0.4.1, 00:29:32

C 10.0.51.4/32 is directly connected, Loopback0

B 192.168.1.0/24 [20/2] via 10.0.4.1, 00:29:55

B 192.168.2.0/24 [20/0] via 10.0.4.1, 00:29:55

192.168.3.0/32 is subnetted, 3 subnets

B 192.168.3.1 [20/3] via 10.0.4.1, 00:29:55

B 192.168.3.2 [20/2] via 10.0.4.1, 00:29:55

B 192.168.3.3 [20/0] via 10.0.4.1, 00:29:55

### Ip route for core switch routers:

**SwitchR1#show ip route**

Gateway of last resort is not set

10.0.0.0/8 is variably subnetted, 10 subnets, 2 masks

C 10.0.1.0/24 is directly connected, FastEthernet1/0/1

L 10.0.1.1/32 is directly connected, FastEthernet1/0/1

C 10.0.2.0/24 is directly connected, FastEthernet1/0/2

L 10.0.2.1/32 is directly connected, FastEthernet1/0/2

O E2 10.0.3.0/24 [110/1] via 192.168.1.1, 00:14:09, GigabitEthernet1/1/2

O E2 10.0.4.0/24 [110/1] via 192.168.1.1, 00:14:08, GigabitEthernet1/1/2

B 10.0.51.1/32 [20/0] via 10.0.1.2, 01:10:59

B 10.0.51.2/32 [20/0] via 10.0.2.2, 01:10:59

O E2 10.0.51.3/32 [110/1] via 192.168.1.1, 00:33:25, GigabitEthernet1/1/2

O E2 10.0.51.4/32 [110/1] via 192.168.1.1, 00:33:18, GigabitEthernet1/1/2

192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks

C 192.168.1.0/24 is directly connected, GigabitEthernet1/1/2

L 192.168.1.2/32 is directly connected, GigabitEthernet1/1/2

O 192.168.2.0/24 [110/2] via 192.168.1.1, 01:11:00, GigabitEthernet1/1/2

192.168.3.0/32 is subnetted, 3 subnets

C 192.168.3.1 is directly connected, Loopback0

O 192.168.3.2 [110/2] via 192.168.1.1, 01:11:00, GigabitEthernet1/1/2

O 192.168.3.3 [110/3] via 192.168.1.1, 01:08:01, GigabitEthernet1/1/2

**SwitchR2Center#show ip route**

Gateway of last resort is not set

10.0.0.0/8 is variably subnetted, 8 subnets, 2 masks

O E2 10.0.1.0/24 [110/1] via 192.168.1.2, 01:01:37, GigabitEthernet1/1/2

O E2 10.0.2.0/24 [110/1] via 192.168.1.2, 01:01:37, GigabitEthernet1/1/2

O E2 10.0.3.0/24 [110/1] via 192.168.2.2, 00:18:28, GigabitEthernet1/1/1

O E2 10.0.4.0/24 [110/1] via 192.168.2.2, 00:18:27, GigabitEthernet1/1/1

O E2 10.0.51.1/32 [110/1] via 192.168.1.2, 01:03:55, GigabitEthernet1/1/2

O E2 10.0.51.2/32 [110/1] via 192.168.1.2, 01:03:55, GigabitEthernet1/1/2

O E2 10.0.51.3/32 [110/1] via 192.168.2.2, 00:37:44, GigabitEthernet1/1/1

O E2 10.0.51.4/32 [110/1] via 192.168.2.2, 00:37:37, GigabitEthernet1/1/1

192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks

C 192.168.1.0/24 is directly connected, GigabitEthernet1/1/2

L 192.168.1.1/32 is directly connected, GigabitEthernet1/1/2

192.168.2.0/24 is variably subnetted, 2 subnets, 2 masks

C 192.168.2.0/24 is directly connected, GigabitEthernet1/1/1

L 192.168.2.1/32 is directly connected, GigabitEthernet1/1/1

192.168.3.0/32 is subnetted, 3 subnets

O 192.168.3.1 [110/2] via 192.168.1.2, 01:15:20, GigabitEthernet1/1/2

C 192.168.3.2 is directly connected, Loopback0

O 192.168.3.3 [110/2] via 192.168.2.2, 01:12:30, GigabitEthernet1/1/1

**SwitchR3#show ip route**

Gateway of last resort is not set

10.0.0.0/8 is variably subnetted, 10 subnets, 2 masks

O E2 10.0.1.0/24 [110/1] via 192.168.2.1, 00:59:45, GigabitEthernet1/1/2

O E2 10.0.2.0/24 [110/1] via 192.168.2.1, 00:59:45, GigabitEthernet1/1/2

C 10.0.3.0/24 is directly connected, FastEthernet1/0/1

L 10.0.3.1/32 is directly connected, FastEthernet1/0/1

C 10.0.4.0/24 is directly connected, FastEthernet1/0/2

L 10.0.4.1/32 is directly connected, FastEthernet1/0/2

O E2 10.0.51.1/32 [110/1] via 192.168.2.1, 01:02:02, GigabitEthernet1/1/2

O E2 10.0.51.2/32 [110/1] via 192.168.2.1, 01:02:02, GigabitEthernet1/1/2

B 10.0.51.3/32 [20/0] via 10.0.3.2, 00:35:52

B 10.0.51.4/32 [20/0] via 10.0.4.2, 00:35:44

O 192.168.1.0/24 [110/2] via 192.168.2.1, 01:10:36, GigabitEthernet1/1/2

192.168.2.0/24 is variably subnetted, 2 subnets, 2 masks

C 192.168.2.0/24 is directly connected, GigabitEthernet1/1/2

L 192.168.2.2/32 is directly connected, GigabitEthernet1/1/2

192.168.3.0/32 is subnetted, 3 subnets

O 192.168.3.1 [110/3] via 192.168.2.1, 01:10:36, GigabitEthernet1/1/2

O 192.168.3.2 [110/2] via 192.168.2.1, 01:10:36, GigabitEthernet1/1/2

C 192.168.3.3 is directly connected, Loopback0

### Neighbors in routing protocols:

**Mar1#show bgp neighbor**

**BGP neighbor is 10.0.1.1, remote AS 65001, external link**

BGP version 4, remote router ID 192.168.3.1

BGP state = Established, up for 01:11:09

Last read 00:00:18, last write 00:00:23, hold time is 180, keepalive interval is 60 seconds

Sent Rcvd

Opens: 1 1

Notifications: 0 0

Updates: 2 13

Keepalives: 79 78

Route Refresh: 0 0

Total: 82 92

Default minimum time between advertisement runs is 30 seconds

…output omitted…

As you can see, the local router only has one eBGP neighbor, SwitchR1

**SwitchR1#show bgp neighbor**

BGP neighbor is 10.0.1.2, remote AS 65011, **external link**

BGP version 4, remote router ID 10.0.51.1

BGP state = Established, up for 01:12:29

…output omitted…

BGP neighbor is 10.0.2.2, remote AS 65012, **external link**

BGP version 4, remote router ID 10.0.51.2

BGP state = Established, up for 01:12:29

…output omitted…

BGP neighbor is 192.168.3.3, remote AS 65001, **internal link**

BGP version 4, remote router ID 192.168.3.3

BGP state = Established, up for 01:08:29

Last read 00:00:42, last write 00:00:00, hold time is 180, keepalive interval is 60 seconds

…output omitted…

SwitchR1, however, has three neighbors: two eBGP local routers, and an iBGP SwitchR3.

**SwitchR2Center#show ip ospf nei**

Neighbor ID Pri State Dead Time Address Interface

192.168.3.3 1 FULL/BDR 00:00:37 192.168.2.2 GigabitEthernet1/1/1

192.168.3.1 1 FULL/BDR 00:00:30 192.168.1.2 GigabitEthernet1/1/2

SwitchR2, the center switch, doesn’t run BGP. Thus, it only has OSPF neighbors.

## Problems encountered

**Thought I must use static binding:**

This lab requires the captured label with labels I configured. So, I spent a lot of time figuring out how to use static binding command:

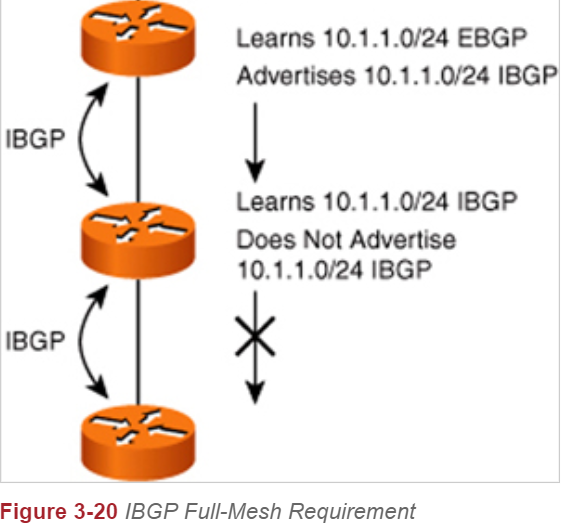
mplsstaticbindingipv4**prefix mask [**input**|** output**nexthop] label**

But instead, the range [lower upper bound] command is enough:

mpls label range 16 1048575

**(Previously) iBGP local routers can’t connect to each other:**

I thought local routers should be in a local network; thus, they should use iBGP. But in fact, iBGP requires a fully-mesh network which my local router don’t have (if I really what to use iGBP, I need to manually configure route reflector to prevent routing loop). Thus, eBGP is more appropriate for this situation.



A link between R1 and R3 makes this a full-mesh network (every knot is directly connected to every other

Without full-mesh network, R1 and R3 don’t establish route to each other.

Thus, can’t use iBGP for links between local routers and PE switch.

Picture credit: [www.ciscopress.com/articles/article.asp?p=1763921&seqNum=7](http://www.ciscopress.com/articles/article.asp?p=1763921&seqNum=7)

**Not seeing the MPLS labels because I used OSPF:**

OSPF is an Interior Route Protocol (in contrast to Border Gateway Protocol). MPLS is intended to run in Large Border Gateway conditions. Thus, even if I enable MPLS, it will become tag-switching, which is not what I intended. I started out configuring OSPF for all routers (core and local ones), but after I realize this problem, I switched from OSPF to BGP. And as I learn more about the difference between eBGP and iBGP, I decide on my final topology, which uses both eBGP and iBGP.

**BGP “mask” command mismatch on two directly connected interfaces:**

Both classful and classless (with mask or without mask command) BGP will work. However, BGP neighbor can’t be established, if two side of the link don’t match each other (classful to classless, or classless to classful).

router bgp 65014

network 10.0.4.0 mask 255.255.255.0

or (both works)

router bgp 65014

network 10.0.4.0

But I didn’t spot the classless/classful mismatch, and it took me a long time to troubleshot the network.

**Failure in Route Redistribution:**

Since I am using two routing protocols, **TWO** route redistribution commands were need. One redistributing BGP routes into OSPF routing table. The “subnets” command enables classful BGP routing (with mask).

redistribute bgp 65001 subnets

IN BGP, redistribute OSPF routes into BGP:

redistribute ospf 1

I didn’t remember to redistribute OSPF into BGP routing table, and it took me a while to troubleshot.

**Monitor Session Command:**

At first, I am not sure which port to monitor (G1/1/2 or Fa1/0/1 on SwitchR1? The correct answer is G1/1/2 because it’s in the MPLS domain). During the beginning of the lab, I have to constantly change monitor session ports, which take considerable about of time.

monitor session 1 source int G1/1/2

monitor session 2 source int fa1/0/1

monitor session 1 destination int fa1/0/23

monitor session 2 destination int fa1/0/24

What I learnt was that I can use a single monitor session to combine the monitoring of G1/1/2 and Fa1/0/1 into one port.

monitor session 1 source int G1/1/2

monitor session 1 source int fa1/0/1

monitor session 1 destination int fa1/0/24

In the first configuration, I have to constantly swap between fa1/0/23 and fa1/0/24 to get traffic from both ports. In the second configuration, I don’t have to swap the monitoring destination ports and that save a lot of time.

## Summary

A big part of this lab is to figure out the mechanism of MPLS and how to implement MPLS on routers. After knowing learning about MPLS, another big part is to decide which routing topology to use (all OSPF network? All BGP network? Both BGP and OSPF? Internal BGP or external BGP?). After a lot of research, I decided to use iBGP within the MPLS core network on the Provider Edge routers. I used iBGP because usually a cooperation get only one BGP AS number (BGP AS number is leased with considerable cost); then, I decided to use OSPF to connect the inner routers within the server provider MPLS core network, because iBGP alone requires full-mesh topology, which SwitchR1, SwitchR2, and 3 don’t have.

Furthermore, since I am not familiar with BGP (both iBGP and eBGP), I ran into multiple routing protocol problems and have to troubleshot them. This lab has been a challenging, yet fun combination of team working with my classmates and researching online documentation on MPLS network.