

MPLS Lab 2 - MPLS VRF VPNS

Lab Review by

Tom Woo
Roland Szczesny
Matt Capranos

Table of Contents

Executive Summary.....page 3

Network Layout.....page 4

Analysis of LDP at work.....page 5

 Initialization message.....page 5

 Address Message – Label mapping message.....page 6

 Label withdraw message.....page 7

Analysis of BGP with extend communities.....page 8

 Extended Communities.....page 9

Configuration.....page 10

Conclusion.....page 13

Executive Summary

The purpose of this lab is to simulate an ISP providing a VPN service to different customers. The negotiation of the services will occur on the Customer – Provider edge of the network, the core of the ISP network will focus on label switching. Within the Customer to Provider VPN service, different protocols were implemented for this environment, the protocols (MPLS, LDP, BGP and of course OSPF) were needed to complement and interact with one another to allow a VPN service.

This lab allowed us to study the interaction among the four different protocols (MPLS, LDP, BGP and OSPF) in order to build a core network to provide a VPN service without having to explicitly establish tunnels. We will focus on the following topics:

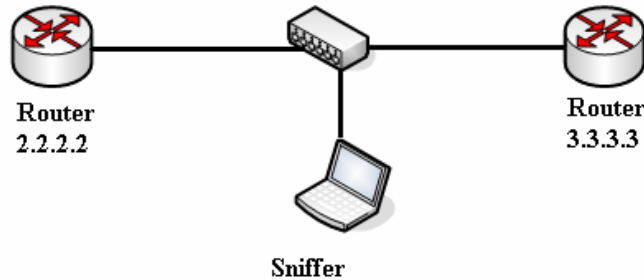
- Detailed sniffed analysis of LDP protocol at work
- Detailed sniffed analysis of BGP with extended communities
- Configuration

Analysis of LDP at work

Within this section we will describe the Label Distribution Protocol at work, LDP is used between MPLS nodes in order to establish and maintain the label bindings. For MPLS to operate correctly, the label distribution information needs to be transmitted reliably, and the label distribution protocol message pertaining to a particular FEC (Forwarding Equivalence Class) need to be transmitted in sequence. We will analyze LDP in three distinct stages, those stages are:

- Initialization message
- Address Message – label mapping message
- Label withdraw message

All of the sniffs were taken between Routers 2 and 3 with use of a hub.



Initialization Message

While sniffing we were able to catch an initialization message being sent from Router 3.3.3.3 to Router 2.2.2.2, and moments later a keep alive message was sent back from Router 2.2.2.2 to Router 3.3.3.3 as show below.

No. -	Time	Source	Destination	Protocol	Info
835	489.97213	3.3.3.3	2.2.2.2	TCP	11006 > 646 [ACK] Seq=1 Ack=1 Win=4128 Len=0
836	489.95383	3.3.3.3	2.2.2.2	LDP	Initialization Message
837	489.95008	3.3.3.3	2.2.2.2	TCP	646 > 11006 [ACK] Seq=27 Ack=37 Win=4092 Len=0
838	489.96776	2.2.2.2	3.3.3.3	LDP	Initialization Message Keep Alive Message
839	489.97031	3.3.3.3	2.2.2.2	TCP	11006 > 646 [ACK] Seq=37 Ack=45 Win=4084 Len=0

The contents of the LDP initialization message and explanation are found on the next page.

MPLS Lab 2 – MPLS VRF VPN

The LDP initialization message was sent from Router 3.3.3.3 to Router 2.2.2.2, the LDP message contains the LDP version number, the Label switching router ID. The initialization message is marked as being an **initialization message**, the next interesting piece of information contained in the message, is in the Parameters field. In the parameters field, the keep alive value is displayed as well as the direction in which the LDP packet is being sent (in this case it is an **Downstream unsolicited proposed**), next would be the expected **session receiver** which is in this case would be receiver LSR 2.2.2.2. An example of the Label Distribution Protocol message is found below.

```
Label Distribution Protocol
  Version: 1
  PDU Length: 32
  LSR ID: 3.3.3.3 (3.3.3.3)
  Label Space ID: 0
  Initialization Message
    0... .... = U bit: Unknown bit not set
    Message Type: Initialization Message (0x200)
    Message Length: 22
    Message ID: 0x0000000d
  Common Session Parameters TLV
    00.. .... = TLV Unknown bits: Known TLV, do not Forward (0x00)
    TLV Type: Common Session Parameters TLV (0x500)
    TLV Length: 14
  Parameters
    Session Protocol Version: 1
    Session KeepAlive Time: 180
    0... .... = Session Label Advertisement Discipline: Downstream Unsolicited proposed
    .0.. .... = Session Loop Detection: Loop Detection Disabled
    Session Path Vector Limit: 0
    Session Max PDU Length: 0
    Session Receiver LSR Identifier: 2.2.2.2 (2.2.2.2)
    Session Receiver Label Space Identifier: 0
```

Address Message - label mapping message

After the initialization messages were exchanged between Routers 3.3.3.3 and 2.2.2.2, a label mapping message was then sent, the address message contains the label mapping message. The label mapping message contains information about the LSR (Router ID) and the link (Link IP address) that are used to form the tunnel from end to end. An example of the label mapping message is shown below, within the message is the Forwarding Equivalence Class, the FEC element is FEC element which matches the packet's destination address.

```
Label Mapping Message
  0... .... = U bit: Unknown bit not set
  Message Type: Label Mapping Message (0x400)
  Message Length: 24
  Message ID: 0x00000014
  Forwarding Equivalence Classes TLV
    00.. .... = TLV Unknown bits: Known TLV, do not Forward (0x00)
    TLV Type: Forwarding Equivalence Classes TLV (0x100)
    TLV Length: 8
  FEC Elements
    FEC Element 1
      FEC Element Type: Prefix FEC (2)
      FEC Element Address Type: IPv4 (1)
      FEC Element Length: 32
      Prefix: 3.3.3.3 ← Router ID
  Generic Label TLV
```

LDP withdraw message

An LSR sends a label withdraw message to its LDP peer to signal that the peer should no longer continue to use specified label that the previous LSR advertised. A Label withdraw message contains the FEC for which the label is being withdrawn, if no label type length value is included in a label withdraw message, all labels that are associated with the FEC are to be withdrawn. Otherwise, only the label that is specified in the label type length value is to be withdrawn. An example of the LDP withdraw message is found below.

```

Label withdrawal Message
0... .... = U bit: Unknown bit not set
Message Type: Label withdrawal Message (0x402)
Message Length: 24
Message ID: 0x00000389
Forwarding Equivalence Classes TLV
00.. .... = TLV Unknown bits: Known TLV, do not Forward (0x00)
TLV Type: Forwarding Equivalence Classes TLV (0x100)
TLV Length: 8
FEC Elements
FEC Element 1
FEC Element Type: Prefix FEC (2)
FEC Element Address Type: IPv4 (1)
FEC Element Length: 30
Prefix: 12.12.12.0
Generic Label TLV
00.. .... = TLV Unknown bits: Known TLV, do not Forward (0x00)
TLV Type: Generic Label TLV (0x200)
TLV Length: 4
Generic Label: 16

```

A LSR that receives a label withdraw message must acknowledge it with a label release message. The LSR also uses label release message to indicate that it no longer needs that specific label that was previously requested by its LDP peer. A Label Release message contains the FEC for which the label is being released. If no Label TLV is included in a Label Release message, all labels that are associated with the FEC are to be released. An example of the release message is found below.

```

Label Release Message
0... .... = U bit: Unknown bit not set
Message Type: Label Release Message (0x403)
Message Length: 24
Message ID: 0x00002c77
Forwarding Equivalence Classes TLV
00.. .... = TLV Unknown bits: Known TLV, do not Forward (0x00)
TLV Type: Forwarding Equivalence Classes TLV (0x100)
TLV Length: 8
FEC Elements
FEC Element 1
FEC Element Type: Prefix FEC (2)
FEC Element Address Type: IPv4 (1)
FEC Element Length: 30
Prefix: 12.12.12.0
Generic Label TLV
00.. .... = TLV Unknown bits: Known TLV, do not Forward (0x00)
TLV Type: Generic Label TLV (0x200)
TLV Length: 4
Generic Label: 16

```

Analysis of BGP with extended communities

In this section we will discuss the extended communities that are being sent within the BGP update messages. Multi-protocol BGP is an extension to BGP-v4 implementation that makes use of the Type, Length Value fields; this allows additional functionality within BGP. Below is a screen showing the additional extensions within a BGP update message between Routers 3.3.3.3 and 1.1.1.1.

```

  ▣ Border Gateway Protocol
    ▣ UPDATE Message
      Marker: 16 bytes
      Length: 115 bytes
      Type: UPDATE Message (2)
      Unfeasible routes length: 0 bytes
      Total path attribute length: 92 bytes
    ▣ Path attributes
      ▣ ORIGIN: INCOMPLETE (4 bytes)
      ▣ AS_PATH: empty (3 bytes)
      ▣ MULTI_EXIT_DISC: 20 (7 bytes)
      ▣ LOCAL_PREF: 100 (7 bytes)
      ▣ EXTENDED_COMMUNITIES: (35 bytes)
      ▣ MP_REACH_NLRI (36 bytes)
  ▣ Border Gateway Protocol

```

The attributes that are being defined by the extensions apply to the NLRI (Network Layer Reachability Information), which are the MP_REACH_NLRI and MP_UNREACH_NLRI. Below is a screen capture of the MP_REACH_NLRI.

```

  ▣ MP_REACH_NLRI (36 bytes)
    ▣ Flags: 0x80 (Optional, Non-transitive, Complete)
      Type code: MP_REACH_NLRI (14)
      Length: 33 bytes
      Address family: IPv4 (1)
      Subsequent address family identifier: Labeled VPN Unicast (128)
    ▣ Next hop network address (12 bytes)
      Next hop: Empty Label Stack RD=0:0 IPv4=3.3.3.3 (12)
      Subnetwork points of attachment: 0
    ▣ Network layer reachability information (16 bytes)
      ▣ Label stack=28 (bottom) RD=65005:12, IPv4=11.11.11.11/32
        MP Reach NLRI Prefix length: 120
        MP Reach NLRI Label stack: 28 (bottom)
        MP Reach NLRI Route Distinguisher: 65005:12
        MP Reach NLRI IPv4 prefix: 11.11.11.11 (11.11.11.11)

```

In the above capture of the MP_REACH_NLRI several areas of interest can be noted, first the packet was being sent from Router 3 to Router 1, this is determined by the Next_Hop being advertised by router 3 indicating that Router 11.11.11.11 is reachable through Router 3. Secondly, a label is being advertised for this NLRI, the label advertisement is 28 and is the bottom label in the stack. Finally, a Route distinguisher is being carried within the prefix, this route distinguisher is 65005:12.

Extended Communities

In this section we will discuss BGP extended communities. Within the BGP message that was being advertised between Router 3 and Router 1, the BGP update message which a screen capture is shown below, the extended communities' flag being advertise is **optional transitive**, the carried extended attributes found within identify the Route target (65005:12), the OSPF domain (0.0.0.2) OSPF Route Type (Area 0.0.0.0, Type: network) and the OSPF Router ID (5.0.0.3). The sniffed packet is found below.

```
  ▫ Path attributes
    ▫ ORIGIN: INCOMPLETE (4 bytes)
    ▫ AS_PATH: empty (3 bytes)
    ▫ MULTI_EXIT_DISC: 66 (7 bytes)
    ▫ LOCAL_PREF: 100 (7 bytes)
    ▫ EXTENDED_COMMUNITIES: (35 bytes)
      ▫ Flags: 0xc0 (Optional, Transitive, Complete)
        1... .... = Optional
        .1.. .... = Transitive
        ..0. .... = Complete
        ...0 .... = Regular length
        Type code: EXTENDED_COMMUNITIES (16)
        Length: 32 bytes
      ▫ Carried Extended communities
        Route Target: 65005:12
        OSPF Domain: 0.0.0.2
        OSPF Route Type: Area: 0.0.0.0, Type: Network, no options
        OSPF Router ID: 5.0.0.3
    ▫ MP_REACH_NLRI (36 bytes)
```

Configuration

We will be using Router 11 for the explanation of the configurations, as this is an end-point in the VPN tunnel, we will be examining VPN12, we will be focusing on only areas that pertain to the establishment of VRF VPN12.

The *ip vrf vpn12* command is used to define the name of the VPN and is also used to associate a route distinguisher with it, which in our case is 65005:12. The command is shown below:

```
ip vrf vpn12
  rd 65002:12
  route-target export 65002:12
  route-target import 65002:12
```

Next, the *ip cef* and *mpls label protocol ldp* are defined in the global configuration mode, this commands are used to stat that Cisco Express forwarding is required to run on this router (Router 11) and the protocol that will be distributing labels will be LDP, these command is shown below:

```
ip cef
mpls label protocol ldp
```

Two loopback interface were specified on the router, the first interface (Loopback0) is used as the Router ID, the second loopback interface (Loopback12) will be used in the configuration of the VRF VPN, this way the BGP process running the peering routers will be able to establish a peering session. The configurations used be the loopback interfaces are shown below:

```
interface Loopback0
  ip address 11.11.11.11 255.255.255.255
!
interface Loopback12
  ip vrf forwarding vpn12
  ip address 12.0.2.1 255.255.255.0
```

Each of the interfaces that will be used in the LDP label path, require *mpls label protocol ldp* and *tag-switching ip* be applied to them below is an example of such from Router 11 fastethernet 0/1 interface:

```
interface FastEthernet0/1
  ip address 10.0.2.3 255.255.255.0
  duplex auto
  speed auto
  mpls label protocol ldp
  tag-switching ip
```

MPLS Lab 2 – MPLS VRF VPN

The next phase in the configuration was to enable the Routing processes on the Router, for this lab OSPF was used as the IGP (internal gateway protocol) internal to the Autonomous systems, BGP will also be implemented on the Routers. We begin by configuring the OSPF routing process, the configuration is shown below:

```
router ospf 1
  router-id 11.11.11.11
  log-adjacency-changes
  network 10.0.2.0 0.0.0.255 area 0
  network 11.11.11.11 0.0.0.0 area 0
  network 101.101.101.0 0.0.0.3 area 0
```

Next, we configure the BGP process on Router 11, the configuration file is shown below:

```
router bgp 65002
  no synchronization
  bgp log-neighbor-changes
  neighbor 7.7.7.7 remote-as 65001
  neighbor 7.7.7.7 ebgp-multihop 10
  neighbor 7.7.7.7 update-source Loopback0
  neighbor 7.7.7.7 send-community extended
  neighbor 9.9.9.9 remote-as 65002
  neighbor 9.9.9.9 update-source Loopback0
  neighbor 9.9.9.9 next-hop-self
  neighbor 10.10.10.10 remote-as 65002
  neighbor 10.10.10.10 update-source Loopback0
  neighbor 10.10.10.10 next-hop-self
  neighbor 113.113.113.2 remote-as 65004
  neighbor 113.113.113.2 next-hop-self
  no auto-summary
```

A quick explanation is in order concerning the *ebgp-multihop* command for BGP neighbor 7.7.7.7. Normally two BGP peers are directly connected; however, when they are separated the *ebgp-multihop* command was used to establish a BGP peering session between AS 65001 and AS 65002, which is crossing several routers. The *update-source loopback0* states the source IP is the loopback defined as loopback0 from the above configuration. The *send-community extended* command instructs BGP to send communities to it's BGP peer 7.7.7.7

The *address-family ipv4 vrf vpn12* command is used to specify a routing session that will use IPv4 prefixes and that will connect with VPN routing and will be forwarded for that VPN, The *redistributed connected* will be used to redistribute the VPN interface connected into BGP, the command is found below:

```
address-family ipv4 vrf vpn12
  redistribute connected
  no auto-summary
  no synchronization
  exit-address-family
```

MPLS Lab 2 – MPLS VRF VPN

The *address-family vpn4* command is used to enable a routing session that will use a VPN and IPv4 prefixes, in our example the BGP neighbor was configured to 7.7.7.7. The *activate* keyword was used to activate the session, the *send-community extended* command is used for BGP to send its neighbor the specified community, finally, the *exit-address-family* command defines the end of the specific address family, the commands are found below:

```
address-family vpnv4
  neighbor 7.7.7.7 activate
  neighbor 7.7.7.7 send-community extended
  no auto-summary
  exit-address-family
```

In order for the configuration on Router 11 to work, additional configurations needed to be made on Router 3, the configuration and explanation is shown below.

```
router ospf 2 vrf vpn1
  log-adjacency-changes
  redistribute bgp 65005 metric-type 1 subnets
  network 5.0.0.3 0.0.0.0 area 0
  network 103.103.103.0 0.0.0.3 area 0
```

The AS border routers for AS65005 a second instance of OSPF was defined for the VRF VPN. The *redistribute bgp 65005 metric-type 1 subnets* command was used in order to for OSPF to redistribute this process of the OSPF metric type 1 subnet into the BGP process.

Conclusion

The purpose of this lab is to simulate an ISP providing a VPN service to different customers. The negotiation of the services occurred on the Customer – Provider edge of the network, the core of the ISP network focused on label switching. Within the Customer to Provider VPN service, different protocols were implemented for this environment, the protocols (MPLS, LDP, BGP and of course OSPF) were needed to complement and interact with one another to allow a VPN service. This lab allowed us to study the interaction among the four different protocols (MPLS, LDP, BGP and OSPF) in order to build a core network to provide a VPN service without having to explicitly establish tunnels.

The following analysis was performed on a practical test that was assigned to Josh Deluca, Ron Fiander, Adam Molnar and Julian Pagno. The configuration files and sniffs were provided by Ron Fiander, and who also assisted in an in-depth analysis of the configuration files.

Additionally, we at Table 3 would like to thank you for an excellent 6 semesters here at Sheridan College, while challenging and stressful at times we enjoyed our time stressing out in the lab, thank you again for a wonderful college experience.