Intra-Area, Inter-Area and Inter-AS Traffic Engineering and Path Selection Evaluation

by

Ahmed Salem Dokali Abukhshim

A capstone project submitted in partial fulfillment of the requirements for the degree of

Master of Science

in
Internetworking

Department of Electrical and Computer Engineering
University of Alberta

Supervior:
Gurpreet (Pete) Nanda, M.Sc. P.Eng.

ABSTRACT

This capstone project focuses on theoretical and laboratory performance analysis and evaluation for Intra-Area, Inter-Area traffic engineering and path selection and L3VPN passing multiple autonomous systems.

In the first section, chapter one to three focus on service a provider network design, capabilities, challenges and other relevant features that SP network must have, such as routing protocol, LDP vs RSVP, and failure detection and IGP convergence time.

In the second section, chapters four and five, a research and theoretical illustration for Intra-area, Inter-area traffic engineering and inter-AS traffic engineering using different path computations and LSP signaling methods. In fact, per area path computation using local traffic engineering database (TED) and end-to-end path computation using path computation element (PCE) has been discussed in details. In addition, three label switch path (LSP) signaling methods (Contiguous LSP, Stitched LSP and Nested LSP) are discussed. Furthermore, Inter-AS L3VPN option A, option B and option C is also covered in this section.

The last section, section three focus on laboratory implementation for Intra-Area and Inter-Area and Inter-As traffic engineering and path selection. In fact, per-domain path computation and PCE end-to-end path computation has been implemented. Furthermore, LSP protection and recovery have been tested and noted. Later in this section, Inter-AS options A , B, and C have been implemented. Finally, the implementations are evaluated and a recommendation is given.

ACKNOWLEDGMENTS

I would like to express my gratitude and appreciation to my supervisor **(Gurpreet (Pete) Nanda, M.Sc. P.Eng)** for his professional assistance and guidance. In fact, the detailed discussions and the guidance provided helped with protocol behavior, test planning, and testing results analysis.

I would like also to thank Professor **(Mike MacGregor, PHD., PEng)** and Mister **(Shahnawaz Mir)** for the unlimited support and assistant not only during this project but also with all program courses.

TABLE OF CONTENTS

Chapter Title **Page**

LIST OF FIGURES

LIST OF Tables

Table No	Title	Page

1.0 CHAPTER ONE

1.1 Introduction:

TCP/IP is the basic standard protocol of the World Wide Web, it is is a set of rules that control the communications among all nodes on the Interne. It is very important to understand the protocol interaction among the nodes involved in the communication process. One of the most important rules for successful communication is that each station must know the layer 3 address (IP) of the other one. For example, let us assume that a station A need to send some data to a station B. Before A can send the data, A must know the Address of station B, and since B is the final destination, we call this address a destination address. The routers all the way to the destination perform Destination-based forwarding, which means the forwarding decision is based on the destination address.

Figure (1.1) illustrate the entire process, since A and B in a different network, A sends the Packet to it is default gateway (R1). After A gets the packet, hop by hop routing starts. In order for R1 to forward the packet to its next hop A must have an entry in its forwarding table for B. If R1 does not have a valid entry then, the packet gets dropped by R1. However, if a valid entry is available then R1 sent the packet to R2 and R2 do exactly the same as R1 and forward the packet to its next hop. These processes keep happening until A's packet toward B, reaches B.

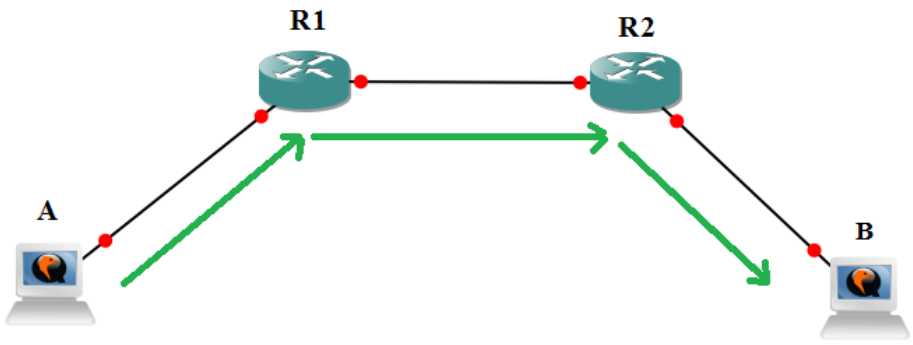


Figure 1.1: TCP/IP Routing Process

The next important thing is how routers build their forwarding tables. There are many techniques to build the routers forwarding table, for example, the routing table can be built statically by using a static route. Static route works by setting static next hop for each possible

destination. The static route in some scenario is the best solution, especially for small networks. Full control of traffic passing the network is one advantage of Static route. However, in large networks static route is not the best solution in term of scalability and fault torarance.

Dynamic routing protocols solve the scalability issue with a static route. There are two types Dynamic routing protocols, Interior Gateway Protocol (IGP) and Exterior Gateway Protocol (EGP). IGP is a dynamic routing protocol, which is used to build the forwarding table within an Autonomous System (AS). In the other hand, EGP is a dynamic routing protocol, which is used to build the forwarding table between Autonomous Systems (AS).

Based on the algorithms IGP uses, IGP is divided into two types. Distance Vector Routing Protocols and Link State Routing Protocols. Choosing which IGP to run on the network depends on the network infrastructure and the network requirements. The next section introduces the differences between them.

1.2 Distance Vector Routing Algorithm vs Link State Routing Algorithm

Distance Vector Routing Algorithm uses distributed computation. As results, each router builds it is forwarding table based on the information received from its directly connected neighbors. Therefore, each router in the network does not have a full overview about entire network. On the other hand, Link State Routing Algorithm (also called shortest-path first) uses replicated distributed database. With Link State Algorithm, all routers within the same network domain have exactly the same database. This database contains all information about the entire network such as all nodes in the network and the cost to each possible destination. Thus, each node in the network calculates its forwarding table based on the database, not as Distance Vector, which builds the forwarding table, based on neighbors' forwarding table.

Link State Routing Algorithm has many advantages ever Distance Vector Routing Algorithm. Table (1.1) summarize the advantages.

	Link State	Distance Vector
Protocols	OSPF, ISIS	RIP, EIGRP
Cost	Metric which reflects the capacity of the links on along the paths	Metric hop counts which is the number of node along the path
Update	Only triggered update after database gets synchronized	Periodically update and triggered an update in case of failure.
Advertisement	Send only link states information	Sends the entire routing table
convergence	Fast	Slow
Loops	Very powerful to prevent loops	Loop is an issue
CIDR and VLSM	Support CIDR and VLSM	Support CIDR and VLSM
Resources	High CPU/memory overhead.	Low CPU/memory overhead.
Implementation	Hard to Implement	Simple to implement
MPLS-TE	Supports traffic engineering	Does not support traffic engineering
Hop count	No limit	Maximum of 15 hops

Table (1.1): Link State vs. Distance Vector

1.3 Open Shortest Path First

OSPF is a links state protocol, it is also part of IP layer; it used protocol number 89 as its protocol identifier. OSPF operate by using a different type of packets, each packet designed for only one specific function. The following paragraphs illustrate OSPF packet and their functions:

Hello packets: hello packets used to identify OSPF neighbors and to maintain neighbors' relationship. It is an auto-discovery mechanism. These packets are multicast periodically to 224.0.0.5 address in point to point link. In broadcast segment, both 224.0.0.5 and 222.0.0.6 multicast address are used. In fact, 224.0.0.6 used to send a hello to DR and BDR and 224.0.0.5 used to send a hello packet from DR and BDR.

Database Description Packets: These packets are exchanged after two routers form a neighbor relationship. The function of this packet is to inform neighbors about the summary of complete topology database contents.

Link State request packets: after the router receives Database Description Packet, it compares its local database with the received database. If there is any new information that is not in the local database or the local information is old. It sends Link State request and keeps sending until the database is synchronized.

Link State update Packets: after the router sends Link State request to its neighbor, the neighbor replies with State update Packet which has all information about the requested Link State.

Link State acknowledgment packets: If the router updates its local database with the new Link State information, it sends the Link State acknowledgment to its neighbor to inform it that I have received your Link State update.

After all, messages exchanged and the local database is updated, the router put itself as the root of the topology and run SPF algorithm in order to find the shortest path to each possible destination. After SPF finishes the calculation, the forwarding table is updated with new information. As long as the network is stable, the forwarding table is still the same and SPF will never run. However, if there is any change in the network, triggered update is sent to inform other routers about this change and in this case, SPF must run again.

1.3.1 OSPF Hierarchical Routing

In a large network, single OSPF area has an issue with scalability. When the network grows, the link state database grows as well. The scalability challenge can be summarized in tree points. the First point, maintaining large database consumes more RAM, and running Dijkstra SPF on large database requires high CPU utilization. The second point, large OSPF database LSAs request and LSAs update take more time, and run SPF on a large database will also take longer time, which leads to longer convergence time. The final point, when the network grows, the chances for more change in the database will also increase. With each change in the database, LSA is flooded and SPF will run in all networks nodes. It is also important to consider flapping links in the network, imagine one of the links goes up and down very fast. With each change, new flooding and new SPF calculation are required.

In order to provide scalability, OSPF designers came with an idea to divide the OSPF network to multiple areas, where each area is independent of other areas. In fact, each area has a separate database and if new links/routes are added in one area the other area is not affected. This solves the scalability issue, but it also introduces new issues such as how two or more area exchange packets and routes and loop issues?

The first OSPF multi-area rule is to design hierarchical topology. In this technique, all OSPF area must connect physical and in some scenarios logically to area 0, area 0 acting as the backbone or transit for entire OSPF domain. The reason behind connecting all areas to area zero is to eliminate routing loops between all areas. This technique is also called, two-level hierarchical topology.

This architecture raises a question, how routes reachability exchanged between areas?. The simple answer to this question is that the routers whose connecting area 0 to other areas responsible for exchanging reachability information. In fact, there are four types of routers in OSPF architecture, internal router, Area Border Router (ABR) and Autonomous System Border Router (ASBR). The following paragraphs summarize the function of each router types:

Internal Router: The function of this router is to maintain link state database for only the area it belongs. It has no knowledge of other areas topologies. All Internal Routers are interconnected to other routers in the same OSPF area.

Area Border Router: Also known as ABR, it connects one or more area to the backbone area(Area 0). This router maintains link state database for backbone area and any other area connected to this router. The main function is to exchange reachability information between its areas.

Autonomous System Border Router: ASBR is the gateway for OSPF domain, this router connected between an OSPF area and any other domain or Autonomous system.

1.3.2 Building OSPF Link State Database

OSPF used different types of links state advertisements to build links state database the following points describes the functions of all LSAs:

- Router LSA, Type 1: Router link advertised by every router within the area it belongs to, type one LSA describe the state of the router links. In fact, this LSA announces the originator presence and carry a list of all the directly connected links within the area.

- Network LSA, Type 2: This LSA only generated within multi-access network segment broadcast and non-broadcast to describe the routers that are connected to that segment. Type 2 LSAs are flooded across their own area only.
- Summary LSA, Type 3: Type 3 LSAs are generated by Area Border Routers (ABRs), type 3 LSAs are advertised from one area to the rest of the areas in the domain.
- ASBR Summary LSA, Type 4: ABR generates a summary ASBR LSAs, which include the router ID of the ASBR in the, so other routers find the ASBR.
- Autonomous System LSA, Type 5: LSA type 5 contains information injected into OSPF from other AS/Domain.
- Group Membership LSA, Type 6: This type of LSA is used for multicast applications.
- Not-so-stubby area LSA , Type 7: Because LSA type 5 are not allowed in Not-So-Stubby-Area (NSSA), type 7 LSAs are used to inject external routes instead of type 5. This LSA is generated by NSSA ASBR and is translated into type 5 LSA as it leaves the NSSA.
- Link-LSAs, Type 8: OSPFv3 uses This LSA. Type 8 LSA is used to give information about link-local addresses and a list of IPv6 addresses on the link.

Each node stores the LSA information In Link State Database, then the router but itself as the root of the tree and calculates the shortest path to each possible destination by running SPF algorithm.

The following Figure (2.1) illustrates how the above LSAs are generated and flooded within OSPF domain.

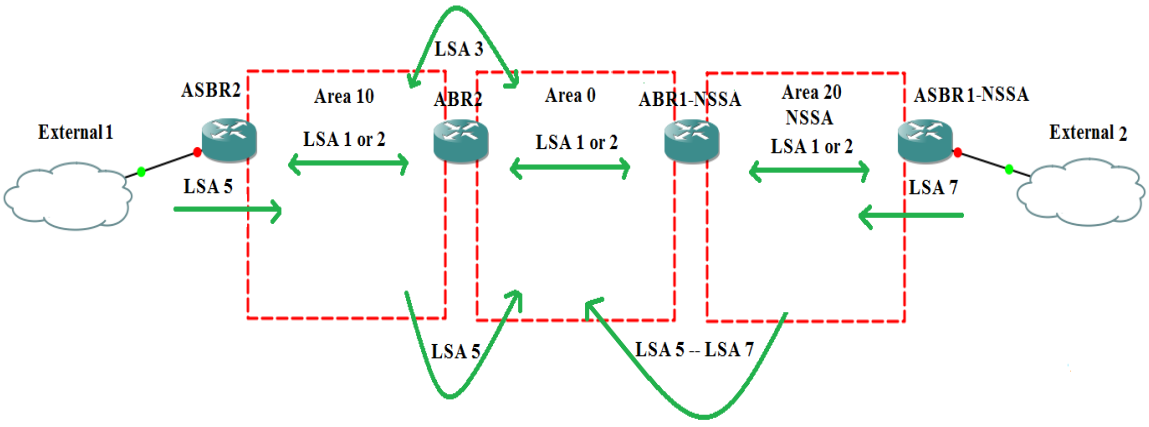


Figure 1.2: LSAs Generation and Distribution

From the above figure, it is clear that LSA 1 and LSA2 are generated within the one area and they are not allowed to leave the area. Instead, LSA 3 is exchanged between backbone area and other areas via ABR. ABR is responsible for Summary LSA 3 generation and distribution.

In NSSA, LSA 5 and LSA 3 are not allowed to be advertised to it. Thus, ABR1-NSSA will block these LSAs. However, it is only possible to advertise LSA 3 default route.

Area 10 is normal OSPF area, thus, ASBR2 generates LSA 5 for the external routes. This LSA 5 is passing through all OSPF except NSSA. In addition, ABR is also responsible for reporting ASBR IP address, this is done by using LSA type 4.

ASBR1-NNSA uses LSA types 7 instead of LSA types 5; this is because LSA 5 is not allowed in this area. Type 7 LSA that contains information about the external routes is advertised by the ASBR to all nodes within the same area. However, When LSA 7 reaches an ABR, The ABR translate LSA 7 back to LSA 5.

1.3.3 OSPF Special Areas

The reason behind special areas is to provide more scalability to OSPF and save resources such as router CPU/RAM. By now, it is known that each area must forward all Inter-area traffic to the backbone area. Thus, backbone area must all information about all other areas.

In many cases, other areas do not need all information about inter-area routes, one default route will get the job done especially when there is only one ABR is used to connect to the backbone area. Thus, LSA type 3, 4 and 5 can be eliminated and replaced with one LSA 3 default route. The following points explain special areas characteristic.

- Stub Areas: this area does not accept external LSA type 5 and 4, LSA 3 default route can be used instead.
- Totally Stubby Area: this area does not accept external LSA type 5 and 4 and inter-area LSA type 3, LSA 3 default route can be used instead.
- Not-So-Stubby Areas: this area does not accept external LSA type 5 and 4, LSA 3 default route can be used instead. However, this area can be connected with external AS/Domain, for this purpose LSA type 7 is used to replace LSA type 5.

- NSSA Totally Stubby Areas: this area is similar to Totally Stubby Area. LSA type 3, 4 and 5 are not allowed. However, this area can be connected with external AS/Domain, for this purpose LSA type 7 is used to replace LSA type 5.

1.3.4 OSPF Convergence

Network convergence is the total time required by the network elements (Hardware and Software) to get the network nodes synchronized and update the forwarding tables after a change in the network topology.

OSPF convergence time depends on many factors as described in the following points:

- Failure Detection Time, it is the time for the network to detect that failure happened, such as Interface failure or node failure. For fast re-convergence, the network must have the capability to detect the failure as it happens. Normally, the time needed to detect that an interface is down relying on interface's Keepalive messages. The estimated failure detection time, in this case, is multi-seconds. However, in the case of L2/L1 device between two OSPF neighbors, keepalive cannot detect the failure at the other end because Keepalive is exchanged between directly connected interfaces. Thus, detecting OSPF neighbor's failure only possible by using OSPF Hello timers. Similar to the first case, failure detection time can reach multiple seconds. One way to improve OSPF failure detection is by using fast hello messages, fast hello decrease the detection time to one second, but OSPF fast hello has one significant disadvantage, all hello packets are processed by the main CPU. Thus, as the number of OSPF neighbors increase the router CPU utilization increases. This badly affects the performance of router control plan. Instead, bi-directional forwarding detection (BFD) can be used. BFD works the same way as keepalive messages. BFD provides low overhead and fast failure detection between two adjacent routers. The main advantage is that BFD implemented in distributed router interface line-cards, thus, saving the control-plane and central CPU from over-utilization. BFD can detect a failure in sub-second and once BFD detected the failure, OSPF is informed about this failure.
- Event Propagation Time: After the router has detected the failure, OSPF sends triggered LSA to its OSPF neighbor. This LSA generation delay has a significant impact on network

convergence time. The generation delay should be as minimum as possible, Cisco recommends 5-10 milliseconds. Another important factor that affects the propagation delay is LSA reception delay. When a router receives LSA on an interface, the interface might be congested, but control traffic has higher QoS priority that other type of traffic. Therefore, LSA reception delay is not an issue. Processing Delay is accounted for the time needed to process the LSA and flood it to OSPF neighbors. Flooding LSA can be done before or after processing it. For fast convergence, fast LSA flood must be enabled on the router. As results, once the router receives OSPF LSA, it advertises it immediately to its OSPF neighbors.

- Packet Propagation Delay: this is the time needed for LSA to reach the destination. Normally, the propagation depends on the physical medium. For example, satellite link has large propagation delay, whereas, the optical fiber has mush lower propagation delay.
- SPF Calculations: Once the router receives LSA, SPF algorithm re-calculation is needed to get the OSPF database updated and synchronized. SPF calculation depends on the speed of CPU.
- RIB/FIB Update: After completing SPF computation, OSPF updates RIB to reflect the changed topology.

As explained above OSPF convergence time depends on many factors, we can summarize all factors in the following formula.

OSPF Convergence Time = Failure Detection Time + Event Propagation Time + SPF RunTime + RIB/FIB Update Time

1.3.5 OSPF Loops and Blackhole

OSPF is very robust against routing loops especially when it comes to intra-area routes. However, during re-convergence, a temporary micro-loop might be present in the network because of inconsistency in the routing table in routers. In details, the first routers detect the failure synchronize their database and update their forwarding table before the other routers. For example, let consider the next topology.

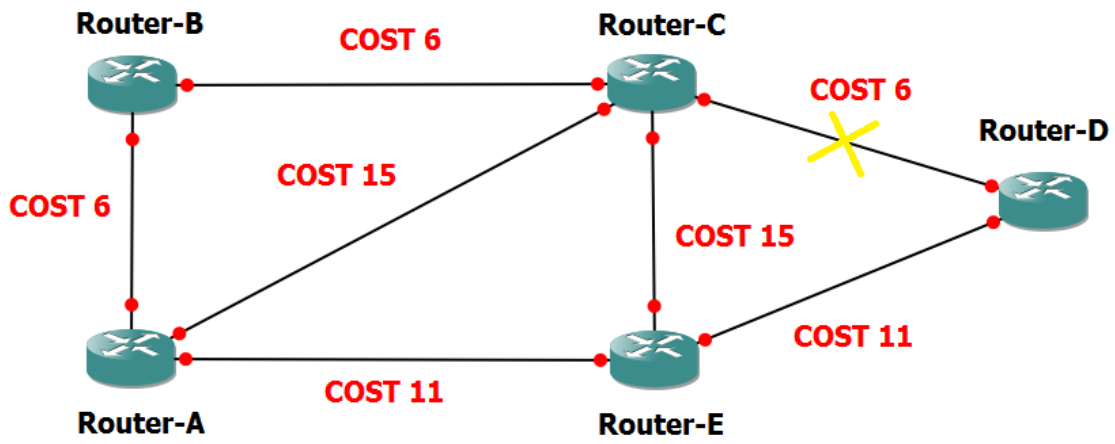


Figure 1.3: Micro-Loops

Let us discuss the routing table of Router-A and B toward Router-D before the link between Router-C and Router D fails. Router-A best path toward D is A-B-C-D because is has the lowest metric and Router-B uses B-C-D path to reach Router-D. Now let us see what will happen if the link fails? Both Router-C and D originate new LSA for that failure to their neighbors. Eventually, all OSPF database get synchronized and the new forwarding table for Router-A toward D will be A-E-D and Router-B uses B-A-E-D.

Let us assume that time difference (LSA origination and Processing) between Router-A and Router-B. If Router-B calculates and installs its new best path through A (B-A-E-D) before A has an opportunity to switch from A-B-C-D to A-E-D, then micro-loop accrues between A and B, each Router A and B pointing to each other as the valid next hop for D. Thus, temporary traffic loss will happen due to this loop.

To eliminate this micro-loop, Loop-Free Alternate Fast Reroute must be enabled in all network nodes. IP-FRR works by precomputation of any alternative path for the same destination. Let us go back to the above scenario and assume that IP-FRR is enabled, in this case, Route- A will have a backup path to toward Router-D ready to be used once the failure is detected. Thus, even if Router-A is going to take a longer time to synchronize its database, micro-loop will be avoided.

It is recommended to delay SPF calculation until all nodes within the area receives the update about the topology change, to do that SPF hold time much be greater than flooding time for within the entire area. Using this method in conjunction with IP-FRR eliminates micro loops, provides fast convergence and saves CPU utilization.

Another type of loops can happen between areas, OSPF treats inter-area routes the same way as distance vector protocol. This is because of the database separation rule between areas. In fact, OSPF behaves as distance vector routing protocol when the ABR exchange routes between areas.

To illustrate this behavior let us analyze the next scenario:

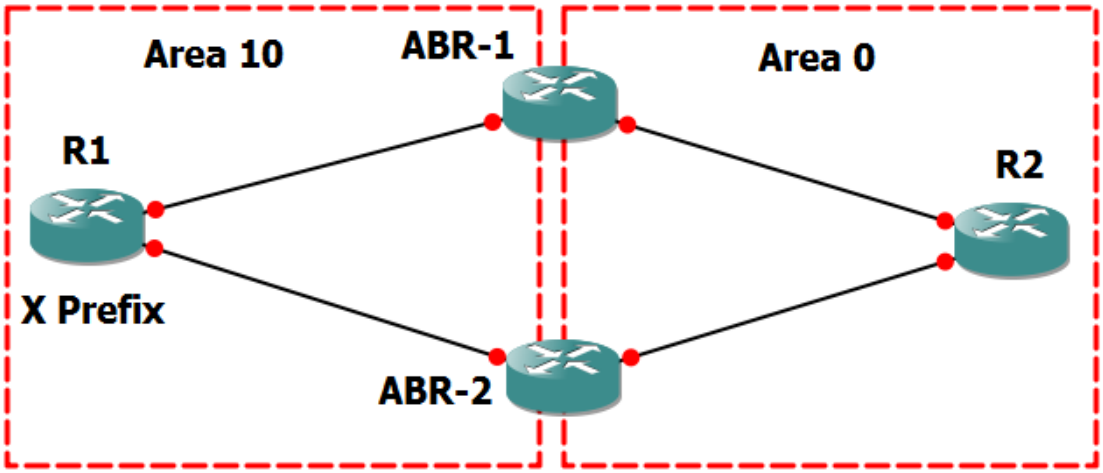


Figure 1.4: OSPF with Distance Vector Behavior

As it shown in figure 4, R1 advertises prefix X in area 10, both ABR-1 and ABR-2 receives X prefix as an Intra-area prefix. In addition, ABR-1 and 2 advertise prefix X to area 0 as summary LSA 3. Let us see what will happen if prefix X fails? the following points explain the convergence process:

- R1 taggers new SLA about the missing route then removes it from OSPF database and run SPF.
- Both ABR-1 and 2 removes the X prefix from area 10 database, but it finds another route to X prefix in area 0. As results, it uses this route and advertises it as summary LSA to area 10. This is the distance vector behavior.
- Because the summary LSA in area 0 depends on area 10, eventually, ABR-1 and 2 removes the summary LSA from area 0.

Another point to notice, it is not only loops will accrues but also the unnecessary SPF calculation that happens at ARBs each time they update each area database.

There are two methods to solve this issue, first, the use inter-area route summarize between areas removes the matching process. Thus, ABR sees the summary LSA as a different prefix. The second method, which is non-standard, OSPF ABR type 3 LSA filtering, stops ABR from advertising area 0 LSA type 3 to other areas.

1.3.6 OSPF Path Selection

OSPF Path selection is determined to follow the next route order, keep in mind that this order is applied to only routes that have the same prefix match, and the router always prepares the longest match regards anything else:

- Intra-Area route is always preferred over any other OSPF routes.
- If there is no match for Intra-Area route, then the router chooses Inter-Area route.
- If Intra and Inter-Area route not available, the next preferable route is External type one.
- If not all the above route types are available, then External type two is selected.
- Finally, the last preferred route is NSSA type 2 if NSSA type 1 is not available.

One important note to remember is that route cost does not take into consideration unless for the same route type, and be aware external and NSSA type 1 route does count the internal metric to the ABSR plus the redistributed metric, whereas, type 2 does not count the internal metric.

2.0 CHAPTER 2

2.1 Introduction

Since the last decade, Service Providers start to deploy Multiprotocol Label Switching (MPLS) in large scale. The main reason choosing MPLS is to increase revenues, but there are several other reasons as well. One of these reasons is that MPLS increases network scalability. For example, in L3-VPN deployment only the edge routers participating in the client control plan and all core routers are unaware of that. The second reason is that using MPLS increases performance. In fact, using label-swapping mechanism eliminates the need for routing lookup. Thus, less CPU and RAM consumes is needed. The third reason is MPLS-based network provides a common platform that supports all type of traffic services.

2.2 MPLS Mechanisms

The primary MPLS property is that it MPLS tunnels different type of traffic over service provider network. The advantage of tunneling technique is that core routers are not aware of tunnel content. In more details, edge router Known as Label Edge Routers (LERs) are responsible for pushing and popping MPLS label, this label assignment at the ingress LER is based on the destination IP address. Thus, ingress LERs must perform L3 lookup to find the next hop for the packet entering the network, then based on next hop, MPLS label is pushed and the packet is forwarded to next router, these type of routers is known as Label Switching Routers (LSRs). In addition, egress LER does the opposite of ingress LER. Egress pops the label and performs L3 lookup to forward the packets to the destination . In addition, the rule of LSRs is label swapping. The MPLS tunnel between two LERs is known as Label Switched Paths (LSPs). LSPs are built between the LERs in order that a packet entering the network at the ingress LER can be transported to the appropriate egress LER. At the LERs, the traffic is classified and mapped to Forwarding Equivalence Class (FEC).

2.3 MPLS Forwarding Plane

MPLS Label as shown in the figure (2.1) consists of many fields the following points describe each field in MPLS header:

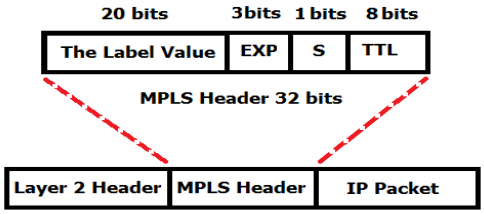


Figure 2.1: MPLS Header Structure

- 20-Bits Label Value : MPLS forwarding is based on this field. Each MPLS enabled Router contains MPLS forwarding table that used to Push, Pop or swap Labels.
- 3-Bits EXP: Experimental Bits are used to distinguish between different types of traffic based on the class of services.
- 1-Bit S: Known as Bottom of stack bit, using this bit to identify whether this label is the last label in the stack or there are more labels. If S bit is equal to one then it means this is the last label in the stack and if it equal to zero then it means there is more label in the stack.
- 8-Bits TTL: time-to-live field used to avoid forwarding loops and it can be using to tracing forwarding path. TTL value is decreased by 1 when it the packet passes each router.

As mentioned above S-bit is used to stack more than one MPLS label in the MPLS header. One MPLS label is sufficient for some services such as transporting public IP traffic. However, many other MPLS services such as Layer 3 VPN, Layer 2 VPN, and VPLS, requires stacking more than one MPLS label. In this case, the ingress LER needs to know which service and to whom this service belongs. To explain the how MPLS packets with two headers are transported between Ingress and Egress LERs, let us analyze the next figure (2.2).

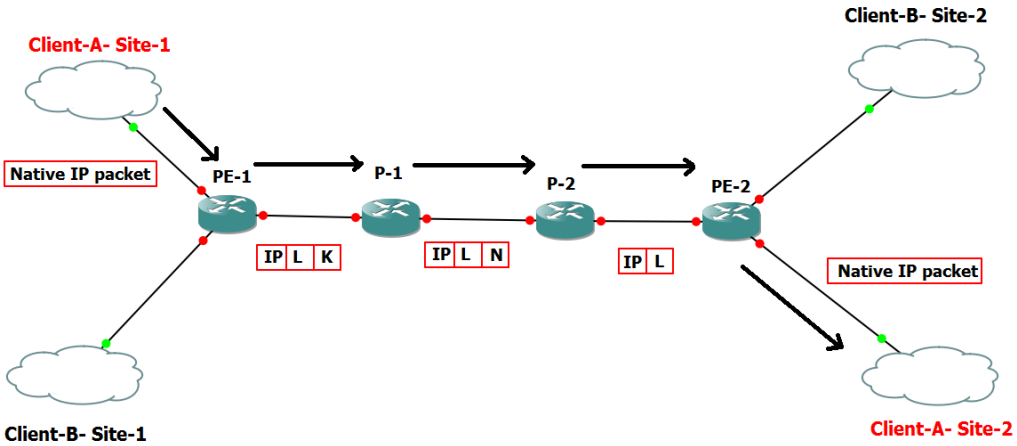


Figure 2.2: MPLS Packet with Two MPLS Labels

As shown above, the inner MPLS label L is used to identify the services and the client instance and the outer header is required to transport the packet over the network from ingress PE-1 to egress PE-2. Client-A has L3-VPN between the two sites, let us analyze an example where client-A-site-1 want to communicate with site-2. In this case, Client send sends IP packet to ingress PE-1. PE-1 performs two lookups one lookup for identifying the service and the second lookup to identify the LSP from ingress PE-1 to egress PE-2. As results, PE-1 end up stacking two MPLS headers (L and K) and forward the packet to P-1. P-1 is a core router, P-1 perform MPLS header lookup and swaps the outer label from K to N and forward the packet to next router P-2.

P-2 is also a core router, but this router has different behavior from PE-1 if Penultimate Hop Popping (PHP) is used. PHP is used to remove the outer MPLS label one hop before its LSP egress destination. PHP helps avoid running out of resources during heavy loads by dividing the workload between the egress LSR and the last core router. Thus, P-2 removes the outer label N and forwards the packet to the egress PE-2.

Once the packet reaches PE-2, there is only one label left in the stack (Label L). PE-2 checks this label and identifies the service and the client associated with it, then it removes it and forwards it to the Client-A-Site-2.

2.4 MPLS Control Plane

After MPLS label forwarding is explained in the previous section, it is important to understand how MPLS label is assigned and distributed in the network. Two main protocols can be used to distribute label binding, Label Distribution Protocol (LDP) and extended Resource Reservation Protocol (RSVP). In addition, Border Gateway Protocol (BGP) can also be used for the same purpose.

2.4.1 Label Distribution Protocol (LDP)

LDP designed for only one purpose that is label-binding distribution throughout the network. LDP uses dynamic peer discovery, the multicast address 224.0.0.2 is used to exchange hellos with neighbors, this hello uses UDP port 646 to discovery neighbors. It is also possible to use unicast messages instead of multicast if the neighbor is not directly connected, and establishing

remote LDP session is required. In this case, it is clear that LDP relies on IGP to establish LDP session.

After the LDP peer is discovered, TCP connection is established between LDP peers and the session is started. Once the LDP TCP connection session is established, the LDP peers negotiate the session parameters that will be used to exchange labels binding; there are two methods to achieve that:

- **Downstream on Demand**: LDP peer advertises label bindings to another peer only if the peer asks for them. This method is undesired in most implementation due to traffic backholed in the network during label requesting process.
- **Downstream Unsolicited**: LDP peer advertises label bindings to peers without being asked. The most implementation uses this method because labels are ready to use at the moment of the packet arrival.

After the label is exchanged, each LDP peer uses the received labels each with it is corresponding FECs when it sends the MPLS packet upstream to that peer.

2.4.1.1 Label Retention and Label Distribution

LDP is the protocol used to distribute binding between FECs and Labels to build MPLS forwarding table that has a mapping between ingress labels and egress labels. MPLS uses downstream label assignment, this means that each LSR is expected to receive MPLS packets with labels that have been generated locally by the LSR. To clarify this let us analyze the next Figure (2.3).

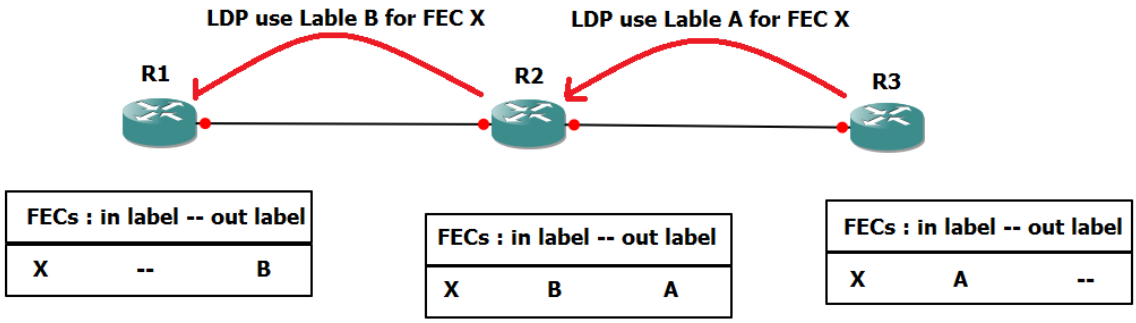


Figure 2.3: LDP downstream label assignment

As shown above, R2 receives label A for FEC X from R3, then R2 generate local label B for that FEC and sent label B to R1. Thus, when R1 has a packet for FEC X, R1 push label B and send it toward R2, Once the packet arrives at R2, R2 checks it is local MPLS forwarding and swap B label and A label, this process will take a place until the packet arrives at the end of LSP.

The next important question, which labels retention mode, is better to use. There are two modes can be used, conservative and liberal retention modes. Conservative mode refers to maintain only the labels that being used. Whereas, liberal mode keeps all labels ready for use. The liberal mode is the most desired mode because all labels are available for used. However, if the LSR has a limitation in which the number of labels is limited then the conservative mode is the only solution.

One important benefit for choosing liberal retention mode is explained in the next Figure(2.4):

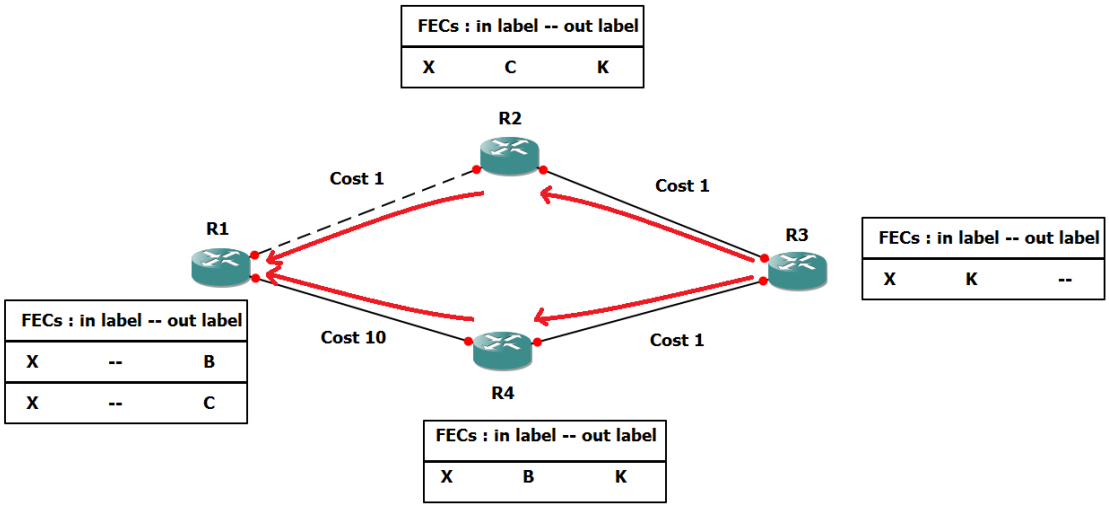


Figure 2.4: Liberal Retention Mode

As shown in the above diagram R3 advertise label K for FEC X to both R2 and R4. R2 and R4 generate local labels for FEC X and advertise them to R1. As results, R1 has two labels for FEC X, label B, and label C and because LDP always follows IGP, R1 will use label C for FEC X. However, if the link between R1 and R2 fails, R1 immediately uses label B after IGP is converged. Therefore, liberal retention mode is very powerful to provide fast recovery assuming IP-FRR is enabled.

2.4.1.2 Ordered and Independent Label Distribution Modes

LDP uses two methods to establish and maintain end-to-end LSP for each FECs. If the ordered control is used then each LSR in the network must check if the LSP who is advertising the label for particular FEC is also the IGP next hop for that FEC. If the check is successful, then the local LSR assign a local label for that FEC and update its MPLS forwarding table then advertise the locally generated label to the upstream LDP peer. This way, ordered control labels distribution ensures that end-to-end LSP creation.

However, with Independent label distribution mode, each LSR generate local labels for each FEC and does not check if the downstream LSR in the best IGP next hop for particular FEC.

To explain the effect of both modes let us consider figure (2.5) and let us assume that all link are active and LDP is enabled on all interfaces except the link between R1 and R2.

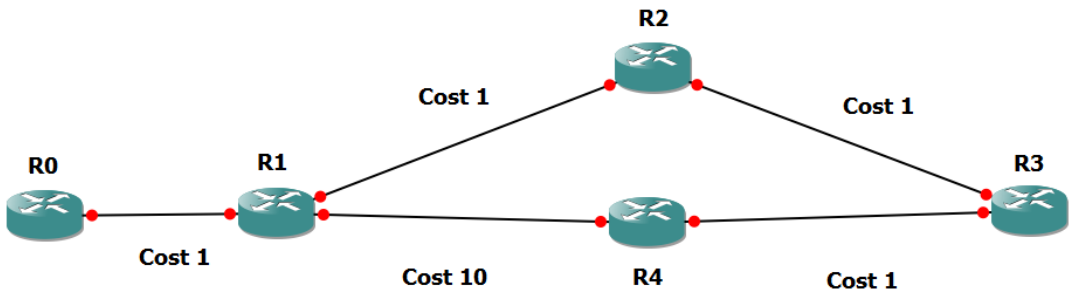


Figure 2.5: Liberal Retention Mode

If ordered control is used then the end-to-end LSP will fail at R0 because R1 will receive a label binding B from R4, and once R1 checks the best IGP path, it will find that the best next hop is R2, but the label is learned from R4. Thus, R1 will not install the label binding in MPLS forwarding table and will not advertise it to any other peer including R0. However, if Independent mode is configured then R1 will not install the label received from R4 because R4 is not the best IGP next hop, but R0 already has a label for this LSP received from R1. Thus, LSP path will fail at R1.

2.4.1.3 LDP and IGP Loss of Synchronization

By now, it is clear that LDP always follows IGP best path. Therefore, LDP and IGP must be always synchronization to prevent traffic loss in the network. Loss of synchronization accrues after IGP

re-convergence. If IGP convergence time is less that LDP convergence, traffic will be lost due to loss of synchronization between IGP and LDP. Let us consider the previous figure (2.5) to analyze this issue:

First, let us assume the following, the link between R1 and R2 is disabled and LSP is needed to be established between R0 and R3. As results, LSP will use R0-R1-R4-R3 path. Now let us assume that link between R1 and R2 is activated. As results of this change, the network will immediately re-converge to use R0-R2-R3, but the LDP take more time to establish the session and exchange labels. Thus, the LSP is broken and the R1 will keep dropping packet until it receives new label assignment.

Two possible ways to solve this issue, the first one is to increase IGP cost for any links with LDP session status down. Thus, the LSP will use any alternative path. The second solution is to use targeted LDP session, this way once the best path is converged the LDP session is ready to for use. Targeted LDP is very helpful especially in the case of a flapping link.

2.4.2 Resource Reservation Protocol (RSVP)

Originally, RSVP was invented to provide a mechanism for bandwidth reservation along the path from the source to the destination. Later, RSVP is improved to be used as a method for distributing MPLS labels. MPLS uses extended RSVP to create and maintain LSPs with some constraints reservation such as bandwidth from the head-end to the tail-end. Each LSP created by RSVP can carry all type of traffic end to end. Since RSVP uses constraints to create LSPs, the LSP can be configured not to follow IGP best path. Thus, RSVP gives the ability to route the traffic away from congested or/and slow nodes. In fact, the head-end can explicitly specify the entire path that LSP must follow, or it can define some nodes along the path that LSP must pass through.

2.4.2.1 RSVP LSPs Setup and Signaling

The ingress LER is responsible for end-to-end LSP. This head-end router sends an RSVP Path message to the egress LER. In addition, the transit LSRs have the ability to inspect this message to make any crucial adjustments. The following points illustrate the contents of Path message:

- Label Request Object (LRO): used to make a request for MPLS labels at each node along the LSP. As results, each router prepares ingress and egress MPLS label for each LSPs.
- Explicit Route Object (ERO): this message used to explicitly specify some or all nodes IP address in which the LSP must pass.
- Record Route Object (RRO): RRO is used to detect loops, each node along the path adds their IP address to RRO list. Therefore, a router will be able to detect a loop if it sees its IP address in RRO list.
- Sender TSpec: this message used to make bandwidth reservation along the path.

Once the tail-end receives Path message, it replies with Resv message. This message is transmitted per hop basis. This means that the tail-end send it to the upstream node and the upstream node sends it to its upstream, this process is repeated until the Resv message arrives to head-end router. The next point and figure (2.6) explain this interaction in details:

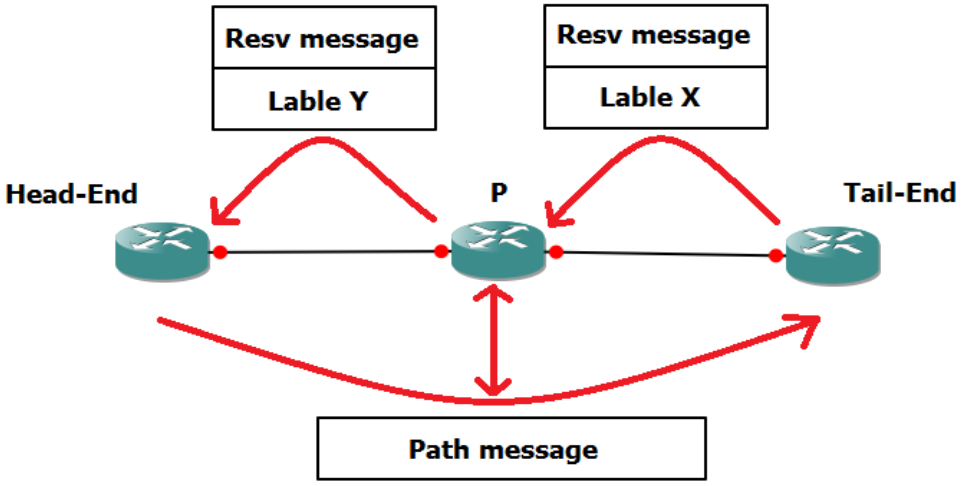


Figure 2.6: RSVP Path and Resv Message Interaction

- Label Object (LO): once the tail-end receives Path message, it replies to an upstream neighbor with Label Object. In the above example, the tail-end informs router P to use label X for any downstream LSP traffic forwarding. Then Router P replaces the label in Resv message with Y, and send this message to its upstream router which is the head-end. Therefore, the head-end used label Y to send traffic to P router, then the P router swaps the Y label with X and forwards it to the tail-end.

- Record Route Object (RRO):RRO has the same function as in the Path message it is used to detect loops.

Once the LSP is successfully created, LSP uses periodic refresh messages to maintain the LSP. Path and Resv messages can be used to achieve that. if a router does not receive a certain amount of these messages, it removes all bandwidth reservation and forwarding states for that LSP. As you can see this method is not scalable because as the number of LSPs increases the overhead increases as well. Therefore, a solution that is more scalable is required, for this purpose Summary Refresh Extension is being used to allow multiple LSPs sessions to refresh their states single message.

In addition, RSVP uses hello message between directly connected RSVP neighbors, the need for this hello is to provide a fast mechanism for failure detection, and it is obvious that depending on RSVP message to timeout it may take longer time.

2.4.3 RSVP and LDP Design Consideration

Both RSVP and LDP must maintain certain number sessions. RSVP sessions are equal to the number of LSPs that passes the routers. In fully meshed RSVP deployment, the number of sessions in rapidly increased. In fact, if we have K number of edge routers (PEs), then the total number of sessions is equal to K-squared. Whereas, in LDP the total number of session in equal to the number of routers.

Furthermore, both protocols must maintain the state of each session. LDP sends periodic keepalives and hellos to a limited number of neighbors. In RSVP, however, each LSP must be refreshed in all nodes along the path.

AS for the forwarding state, LDP keeps the forwarding state for all FECs in the network including the ones that have Equal-cost multi-path (ECMP). On the other hand, RSVP only maintains the state for only the LSP that passes through it.

The advantage of choosing RSVP is that RSVP supports traffic engineering and fast protection; therefore, if these properties are not required, LDP is the most desired solution. Alternatively, it is possible to use both protocols in the network to get the benefit of both protocols while maintaining the network scalability.

2.4.3.1 LDP-Over-RSVP

The most important advantage of LDP-Over-RSVP is seen in very large networks. Full-meshed RSVP LSPs between PE nodes is not scalable because it consumes a huge amount of router resources CPU/RAM. Instead, dividing your network to run LDP in aggregation layer and RSVP in the core layer, reduce the number of RSVP session and provide a scalable solution for protection and traffic engineering. The next example explains the benefits of LDPoRSVP.

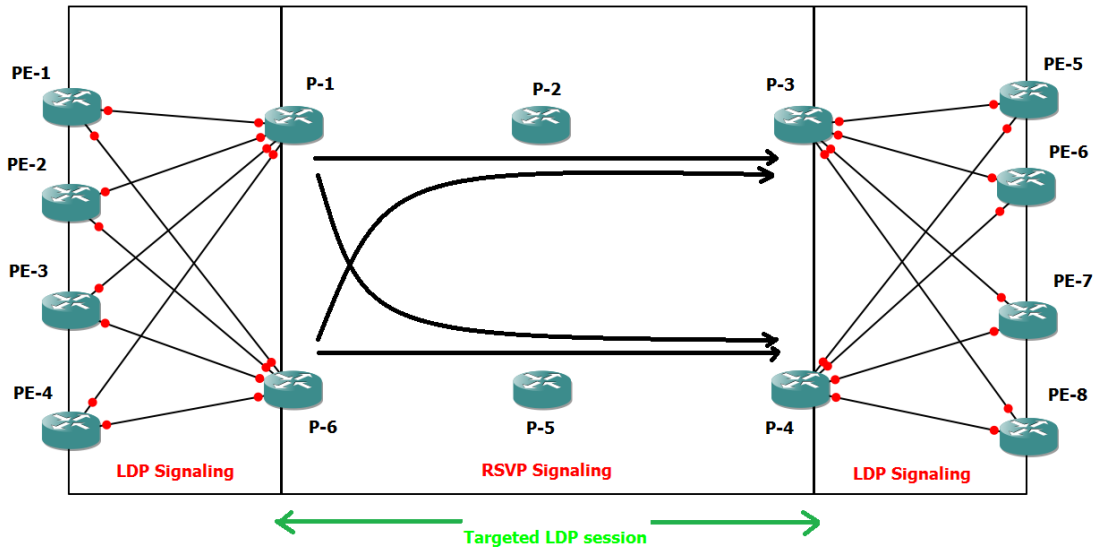


Figure 2.7: LDPoRSVP

As shown in figure (2.7), LDP is used in the aggregation layer among P-1, P-6, P-3, P-4 and PEs sites. However, among P routers RSVP in being used. In addition, targeted LDP session between is also used between P-1, P-6, and P-3, P-4.

Using this scheme eliminates the need for fully meshed LSPs between PEs, and provides a high and scalable solution to provide traffic engineering and protection guarantees among core routers.

End to end communication between PEs is still very effective due to LDP over RSVP implementation. For further illustration of LDP over RSVP let us consider the next example as shown in figure (2.8)

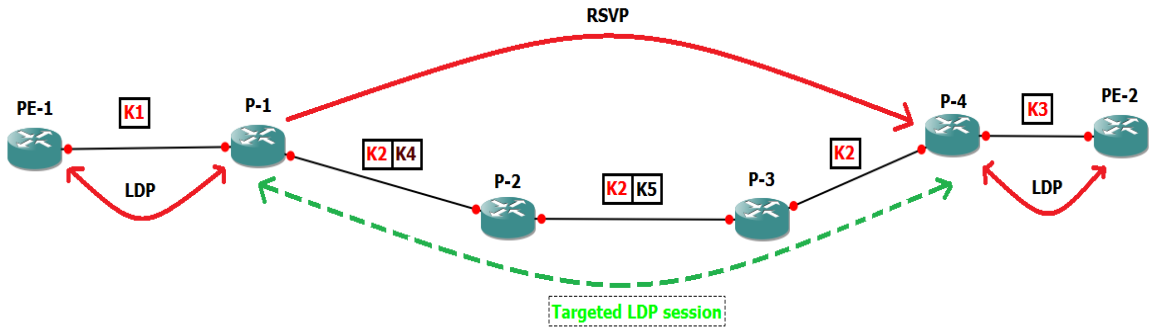


Figure 2.8: LDPoRSVP

LDP is running between edge core router and PEs, however, among core routers, RSVP is running. In addition, there is targeted LDP between P-1 and P-4, in this case, is a necessity to exchange labels between P-1 and P-4 even though P-1 and P-4 are not directly connected.

Let us assume that there is FEC X at router PE-2, by using LDP, PE-2 advertises FEC X binding (K3) to P-4, P-4 install this binding in the forwarding table and generate (K2) a new binding for FEC X and sent it to P-1 by using targeted LDP session. Then P-1 repeats the process and sends FEC X binding (K1) to PE-1. In addition, in this case, RSVP is only been used as LSP tunnel to tunnel the traffic between P-1 and P-4.

After the control plane is clear, let us consider the forwarding plane as shown in the figure, PE-1 has traffic associated with FEC X. PE-1 pushes K1 label learned from LDP then it forward the packet to P-1. P-1 checks MPLS forwarding table and swap K1 label with K2, K2 label is learned from P-1 over targeted LDP session, also, P-1 pushes a second label K4 to tunnel the traffic over RSVP LSP. P-2 only checks the outer label K4, and then it swaps it with K5. If PHP is used, P-3 pops K5 label and send the packet to P-4; P-4 is the end of targeted LDP session. P4 has binding for FEC X in its MPLS forwarding table, thus, P4 swap K2 label with K3 and forwarded to the final destination PE-2.

2.4.3.2 RSVP LSP Hierarchy

In some cases, End-to-End LSPs is required, but still having fully meshed LSP is not scalable in a large network as the number of LSPs exponentially increase. Hierarchy LSPs is a method to reduce the total number of LSPs inside the core LSRs. In fact, End-to-End LSPs are invisible at

core router because all End-to-End LSPs are nested in core LSPs. As result, Core LSRs only signal and maintain Core LSPs.

To explain how LSP hierarchy works let us consider the next example, as shown in the next figure (2.9)

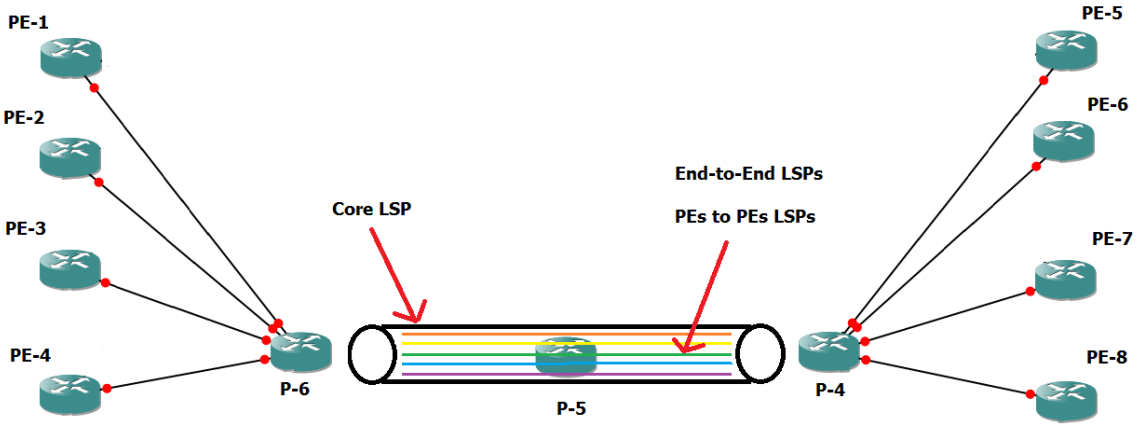


Figure 2.9: LSP Hierarchy

As shown above, each PE might is fully meshed to other PEs to provide end-to-end LSPs. However, there are additional LSPs in the core; these LSPs are used to tunnel End-to-End LSPs inside the core LSP. This hierarchy LSPs implementation helps to reduce the number of LSPs in the core while maintaining End-to-End LSPs.

The control Plane and forwarding, in this case, is somehow similar to LDPoRSVP, In fact, P-5 does not know any things about PEs LSPs, only P-4 and P-6 are participating in the exchange of control plane information about the End-to-End LSPs. As for the forwarding plane, P-6 will swap the labels and push another label for core LSP and P-5 only swaps the outer label or pop the label if PHP is enabled.

2.4.4 BGP Label Distribution

Multiprotocol-Border Gateway Protocol (MP-BGP) can also be used to distribution MPLS labels across the network. Mainly, MP-BGP is used between PEs to distribution VPNs labels. However, LDP or RSVP is still needed as core transport layer. In fact, MP-BGP mainly used to identify the service and service instance.

MP-BGP labels distribution is also used as a solution for Hierarchical and Inter-As VPNs, across multiple ASs. BGP Label Distribution is discussed in more details in chapter five.

2.5 MPLS-TE - Introduction

MPLS Traffic Engineering (MPLS-TE) has strong capabilities to influence the traffic inside the network. There are several reasons to enable MPLS-TE, first MPLS-TE helps to avoid congested link. For example, without using MPLS-TE, traffic routing will be based on metric and the metric has some limitation when it comes to the current traffic at each node or link. Thus, MPLS-TE can route the traffic away from a congested node or link. In addition, MPLS-TE is also saved the cost by delaying any new physical implementation to increase capacity in the network by splitting the traffic to use other unutilized links. Furthermore, some application such as voice and video application is time sensitive, thus, MPLS-TE also can be used to reroute the traffic into low-latency links.

However, if the network utilization is low, and there is neither high-latency links nor congestion in the network, MPLS-TE may not be required.

2.6 MPLS-TE Traffic-Engineered Paths Calculation and Setup

There is one important step to implement MPLS-TE, which is the path setup, the path must be computed according to the LSP constraints and these constraints must be achieved end-to-end.

2.6.1 Path Constraints and Link Properties Distribution

In order for LSP setup to work, the path must satisfy a set of Constraints when the path is calculated End-to-End. The following points include some of these Constraints:

- Links traffic engineering bandwidth.
- Links color also known as administrative attributes.
- Links traffic engineering metric.
- LSP bandwidth.
- The maximum hops count for each LSP.
- Include or exclude some hops within the network.

Administrative attributes are a way to include or exclude link or links from being used in LSP path. The concept is straightforward links can be marked with colors. Links can have same of different colors; it is also possible for one link to have more than one color. For example, Link with high delay can be marked with red color, and then you can exclude any red color from being part of voice LSP. This provides an easy and strong method to route the traffic over desired links.

2.6.2 LSP Priorities and Preemption

MPLS-TE uses LSP priorities to mark some LSPs to distinguish between LSP. Each LSP has a priority assigned to it in such away a higher priority LSP can take over lower priority LSP. MPLS-TE LSPs can have up to eight priorities level, 7 is the highest but worst priority and zero is the lowest but the best priority. Each LSP has two associated priorities (Setup priority and Hold priority). Hold on priority used for established LSP access to use resources, whereas, setup priority is used after an LSP is established.

Setup and hold priorities are used in conjunction to determine which LSP has the highest priority. For example, if new LSP in setup mode and there is no enough resources, then the new LSP setup priority is compared with the existing LSP hold priority, and if the setup priority is higher that hold priority, the new LSP preempt the existing one.

Assigning setup and hold priorities must be properly chosen to avoid LSP flapping, for example, let us assume that there is existing LSP-A from node X to Y, and new LSP-B wants to establish in the same link. In this case, if LSP-B has setup priority better that LSP-A holds priority, and LSP-A has better setup priority than LSP-B hold priority, then LSP-B preempt LSP-A, and once LSP-A reestablishes then LSB-A preempts LSP-B. As result, this mistake will create an endless preemption between LSPs. To avoid this issue, never use hold priority worst then setup priority for the same LSP.

2.6.3 Path Calculation and Constrained SPF

CSPF is used to calculate the best path toward the egress router. However, yet the path must satisfy an LSP constraints. CSPF uses the traffic-engineering database (TED). In details, first, each link does not satisfy one or more of the constraints is ignored from the topology, two important

constraints are checked, the bandwidth and link color. Then CSPF calculates choose the best path. If there are multiple equal paths, only one path is selected, yet some vendors are capable of loading balancing across ECMP LSPs. It is also noted that each vendor chooses different tiebreak to select one path from other equal cost paths. For example, Juniper uses the following sequence as tiebreak:

- If multiple paths have equal cost, CSPF chooses the path whose last-hop IP address is the same as the LSP's destination.
- If multiple equal-cost paths still exist, selects the one with the lower hops count.
- If several equal-cost paths remain, CSPF can be used to load balancing.

Traffic engineering database is built by Opaque LSAs exchange between MPLS-TE nodes by using IGP extensions. Both link state routing protocols IS-IS and OSPF have been extended to include TE-specific extensions, which enable them to include MPLS-TE constraints in their database advertisement such information as the bandwidth available for use by LSPs and links administrative attributes. This information is kept in traffic engineering database (TED) at each node to be used for path calculation. Therefore, each node within MPLS-TE domain knows the state of each link in the domain.

OSPF uses Opaque LSAs to increase capabilities in order to distribute traffic engineering information in MPLS-TE domain. In fact, OSPF uses the following Opaque LSAs:

- **Type 9 LSA** - link-local scope flooding.
- **Type 10 LSA** - area-local scope flooding.
- **Type 11 LSA** - AS scope flooding.

2.4.4 Path Setup – RSVP Extensions and Admission Control

After the path is locally calculated, the next step is to LSP setup using RSVP-TE. As described before the path is calculated and LSP is signaled at the head-end LER. Explicit Route Object includes the path end-to-end from head-end to tail-end. In details, ERO carry traffic engineering information that nodes along the path must keep track of such as the bandwidth requested by the LSP. In addition, other information that is very important to path setups such as hold and setup priorities must also be carried by ERO.

As results of ERO message, tail-end LSP sends RESV messages to head-end LSP and at each upstream node admission control is performed, the next points explain the need for the admission control:

- LSPs might be computed without using CSPF.
- The state of recourses availability may change after being calculated by CSPF.
- The path calculation might be done based on out of date traffic engineering database information.

If the admission control succeeded then the path is set up at the node and the new information is advertised using IGP extension to another MPLS-TE node, so they update their local TED, and If the resources are not available, then it is higher setup priority preempt lower hold priority. Furthermore, if resources cannot be located then the reservation fails and an error message is sent to the head-end. Then the head-end re-computes the path.

However, while re-computes the path if the TED is not updated, then the head-end might end up using the same path and fail again; to solve this issue some vendors chose the following two solutions:

- The head-end exclude the node that is responsible for admission control failure. However, the head-end is the only node who knows about this exclusion, thus, other head-end might end up in the same scenario.
- Enable Link state advertisement at the node that is responsible for admission control failure. This advertisement helps TED to be updated. This method helps all nodes because all nodes get updated.

In many cases, after LSP is created over a suboptimal path, and head-end gets updated with new information in the TED, the head-end will try to find a more optimal path. And to prevent any traffic loss, MPLS uses make before break method, where the head-end LSP create another LSP over a better path with the same LSP constraints before switching the traffic from the old LSP to the new one. Once the traffic is switched then the old LSP torn down.

This means for a short period of time, the two forwarding state is maintained for new, and old LSP paths. Therefore, twice the resources will be reserved for a small amount of time. Another issue with make before break method is that the new LSP path might use part of the existing

LSP, thus, again twice the resources are needed. To solve this issue, the two LSP paths must share the bandwidth, with the help of RSVP shared explicit (SE) reservation. the two paths can share the same bandwidth.

2.7 Mapping Traffic Using Traffic Engineering Paths

LSPs tunnels are treated as logical interfaces. In fact, similar concept as L3 routing can be performed. Traffic can be mapped into LSPs by using static route or dynamically by using a dynamic routing protocol.

Mapping the traffic using static route is not a scalable solution because it needs manual configuration. It is very hard to implement in large scale. Thus, most implementations make the routing protocol aware of these LSPs tunnels. With routing route the traffic based on the metric, an LSP metric can be the actual path metric or it can configure manually. With dynamic LSP routing, it is required to enforce head-end router to include these LSPs in SPF calculation this method is known as traffic engineering shortcuts or auto-route. it is also possible to advertise LSPs into link state advertisement so other nodes in the network can take advantage of these LSPs.

The next example illustrates one of the cases where statically and dynamically mapping traffic inside an LSP is beneficial:

- A static route to transit autonomous system: let us assume that an AS is connected to two ASs each from at different node. In this case, LSP tunnel between ASBRs is needed to forward the traffic over the local AS to two other ASs. At both ASBRs, the transit traffic is mapped to the other ASBR next hop due to MP-BGP use. MP-BGP gives the ability to exchange label binding for external prefixes even if there are not directly connected Thus, the BGP-free core can be easy implemented and the internal node does not need to know anything about the external routes. This is method provide high scalability solution to the network.
- Including LSPs in LSA advertisements and SPF calculation: it is recommended that LSPs being advertised to other nodes, so all network nodes can include them in SPF

calculation, but in some cases, the traffic is routed over suboptimal. The next example as shown in figure (2.10) clarifies this issue.

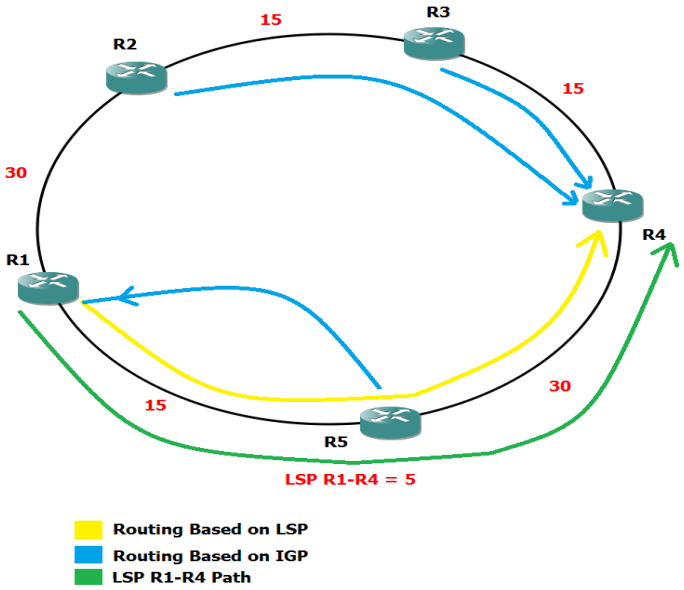


Figure 2.10: Ring Topology and RSVP LSP

R1 has LSP to R4, this LSP is configured to have a metric of 5, this LSP is advertised in LSA, and all nodes include it in their SPF calculation.

Let analyze the results of this process at each node if all nodes have traffic going to R4. Each node will make a decision based on the route metric. R1 sees LSP RA-R4 is the best path and uses the yellow path. However, both R2 and R3 follow the exactly the IGP because it has the lowest metric.

R5 chooses to use the LSP R1-R4, thus, R5 end up uses suboptimal path by sends the traffic to R1, and then R1 maps the traffic to the LSP. This behavior accrues because R5 does not have information about how this LSP is built.

3.0 CHAPTER THREE

3.1 Introduction

Conventionally, Service providers did not provide IP services guarantee, but they provided powerful protection for some services such as voice and L2 services. Traditionally, voice services are provided by dedicated network infrastructure such as Public Switched Telephone Network (PSTN).

To reduce operating expenses (OPEX) and Capital expenditures (CAPEX), service provider networks must evolve from having multiple physical networks for each service to one converged network for all services using TCP/IP model. Therefore, fast service restoration is very important.

One powerful solution that provides fast recovery from failure can be introduced to the physical layer (Layer 1). In fact, SONET/SDH is being used as layer protection mechanism, this solution provides failure recovery time less than 50 ms. SONET/SDH works by maintaining standby link, in which, the services from the working link can be switched to standby link if failure accurses. however, SONET/SDH cost too much because of the unused link and hardware cost to provide the switchover.

Another solution for providing protection is MPLS fast reroute (FRR), MPLS FRR provides comparable protection guarantee as SONET/SDH. FRR works by switching the traffic from one LSP to another LSP. There are two main benefits for using MPLS FRR. First, it provides protection against link and node failure, second, it does not require any additional hardware.

MPLS FR protection traffic loss depends on two factors. First, how fast the failure can be detected? Second, how much time is needed to switch the traffic after the failure is detected.

3.2 Failure Detection

It is very important for the network to have fast failure detection capability. It is possible to have fast failure detection to some level by relying on IGP only. However, the are some drawback, first, IGP fast hello packets can be used to detect a failure within 1 second. Second, in remote failure cases, IGP must flood this information as fast as possible, for this reason, fast LSAs

generation are required. The disadvantage of these techniques is that the extra overhead consumes CPU/RAM resources.

Bidirectional Forwarding Detection (BFD) is a better solution for fast failure detection. BFD is a low-overhead protocol that provides rapid faults detection against L1/L2 failure between two L3 neighbors. The detection time can vary depending on the platform; some platforms can have detection time within 10s of milliseconds. BFD shares the status of the BFD enabled interfaces with IGP. Thus, once the failure is detected the IGP reacts accordingly.

3.3 End-to-End MPLS-TE Protection

One way to provide protection for an LSP is to use path protection. Path protection means that the entire path is protected end-to-end from the head-end to the tail-end. This is done by configuring two LSPs at the head-end, working LSP is known as primary LSP and protection LSP is known as backup LSP.

The backup LSP normally is pre-signaled and ready to take over the traffic once the primary LSP fails at any link/node along the path. Since the head-end is responsible for the switchover then the RSVP error message propagated throughout all nodes until it reaches the head-end. Thus, the switchover time depends on the failure location. In addition, if the secondary LSP is not pre-signaled, then an extra time is needed to get the LSP signaled before the switchover happens. Therefore, the switchover time can be longer.

For successful end-to-end protection implementation, it is very important that primary and secondary LSP do not share any link in the LSP path in order to eliminate any single failure that affects them both. It is also preferable that primary and secondary LSP do not share nodes in the path.

Pre-signaled secondary LSP has one major disadvantage, in many cases, the primary and secondary LSP share the same bandwidth reservation, thus, bandwidth capacity is lost because of unused LSP. To avoid running out of resources the secondary LSP must not be pre-signaled. In few words, it is a trade-off between faster switchover and bandwidth. The following example illustrates when pre-signaled LSP is not recommended.

Let us assume that the network has a shortage in resources and pre-signaled secondary LSP is used. LSP-A primary and secondary are signaled, and then LSP-B needs to get signaled and this LSP using part of the same path LSP-A uses, in this case, if there is a shortage of resources, then primary LSP-B might fail to establish while unused secondary LSP-A is taken the resources. This is a common issue, however, it is possible to solve by manipulating LSPs properties by assigning better setup priorities for all primary LSP, and worse hold priorities for all secondary LSP, to make sure that all primary is established.

3.4 Fast Reroute MPLS-TE Local Protection

This protection scheme is different from end-to-end protection. The idea is simple, do the protection as close as possible to the point of failure. This way, there is no need to wait for RSVP error message to be received by the head-end. Thus, the switchover duration is reduce comparing to end-to-end.

At the head-end, a backup LSP may be configured, but it is not pre-signaled. Once the head-end receives RSVP error message, it will try to re-establish the primary LSP, if the primary LSP failed to establish then it will try to establish the secondary LSP. During this time, the traffic is being forwarded using temporary FRR local protection. To explain how local protection work, let us analyze the next example shown in figure (3.1)

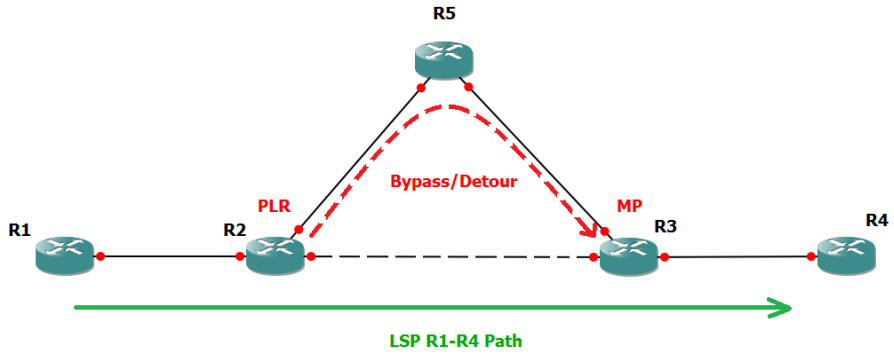


Figure 3.1: MPLS FRR Local Protection

R1 has an LSP to R4 is LSP is working until the link between R2 and R3 fails. R2 will create alternative path known as detour or bypass to go around the failed link. The point creates the detour is known as Point of Local Repair (PLR) and point where the detour is joined the primary

LSP know as Merge Point (MP). This detour will be active until the head-end moves the LSP to another path if there is any.

As mentioned before MPLS known as make-before-break, thus, this detour is created before the failure accrues. The main advantage of local protection is that it has lower packet loss.

Local protection not only resolves link failure but also resolves node failure. In fact, the detour/bypass tunnel can be either one-to-one 1:1 protection or facility N:1 protection. In total, there are four types of different implementations for local failure protection, link protection using 1:1 method, link protection using N:1 method, Node protection using 1:1 method, and lastly, node protection using N:1 method. Each of these four types will be analyzed in the next paragraphs.

3.4.1 1:1 Link Protection

In 1:1 link protection, each LSP have its own detour, this means that the number of detours at each PLR in equal to the number of LSPs passing the PLR. 1:1 protection can also protect the bandwidth requirements. The head-end router can specify whether the LSP is bandwidth protected or not. However, the number of detours might be affected if there are no enough resources at the alternative path. This alternative path is based on TED. In details, the PLR consults the TED to find the best path to each LSP tail-end. Figure (3.2) explains the behavior of 1:1 link protection.

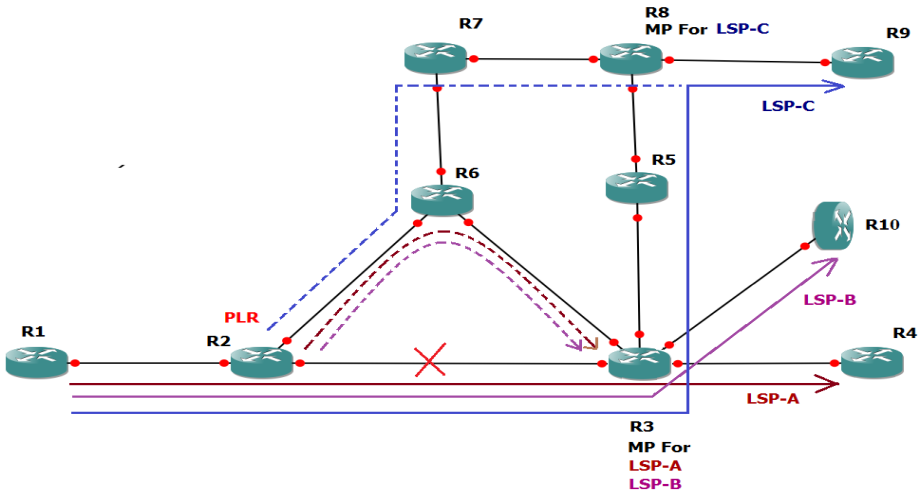


Figure 3.2: MPLS FRR-1:1 Link Protection

It this example, R1 has three LSPs as following: LSP-A tail-end is R4, LSP-B tail-end is R10, and LSP-C tail-end R9. Before link between R2 and R3 fails, LSP-A path is R1-R2-R3-R4, LSP-B path is R1-R2-R3-R10, and LSP-C path is R1-R2-R3-R5-R8-R9.

R2 the PLR, check the TED to find the alternative path for each LSP's tail-end, as results, R3 will be the MP for both LSP-A and B, whereas, LSP-C will use R8 as the MP.

If the link between R2 and R3 fails, then LSP-A will take R1-R2-R6-R3-R4 path, LSP-B will take R1-R2-R6-R3-R10 path, and LSP-C will take R1-R2-R6-R7-R8-R9.

3.4.2 N:1 Link Facility Protection

Facility Protection is a bit different from 1:1 protection, as discussed above in 1:1 protection each LSP will have dedicated detour that created based on the best IGP path to tail-end. on the other hand, facility protection creates only one protection tunnel around the failure. This tunnel will be shared among all LSPs. Therefore, N:1 protection will create the tunnel based on the point of failure IP address. Then at MP each LSP will join the original LSP. One major issue of using this type of protection is that some LSPs might end up using a suboptimal path as it shown in the next example.

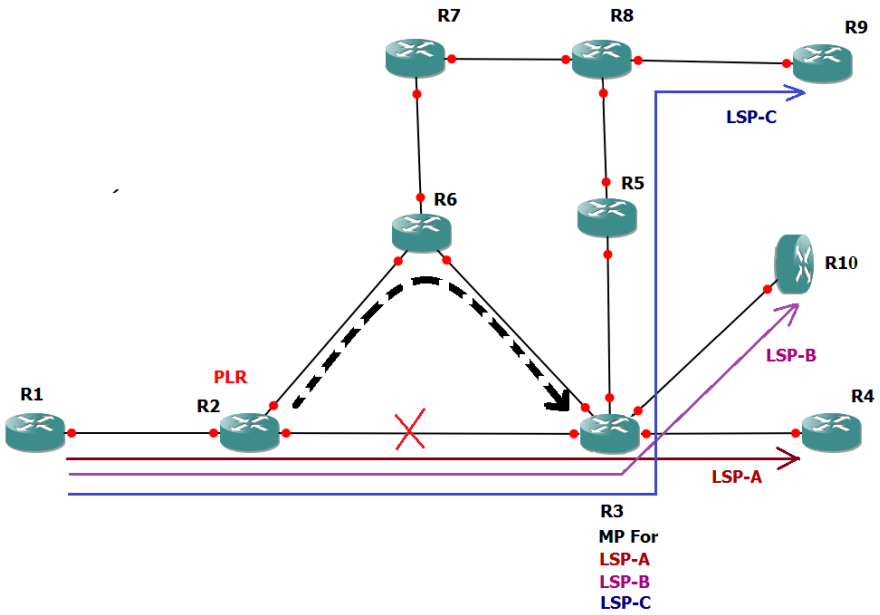


Figure 3.3: MPLS FRR N:1 Link Facility Protection

Before the failure in the link between R2 and R3, LSPs are taking the same path as the previous example with 1:1 protection, but when the failure accrues, the behavior is different. R2 as the PLR consults the IGP to find an alternative best path toward the LSPs next hop. All LSPs uses R3 as next hop will end up using R3 as the MP. Therefore, LSP-A, LSP-B and LSP-C MP is R3. Having this rule may end up using non-optimal path as shown in the example with LSP-C. LSP-C after failure will use R1-R2-R6-R3-R5-R8-R9.

To sum up, choosing between 1:1 protection and N:1 protection is a trade-off between having fewer detours and the chance of having suboptimal paths in the network.

3.4.3 1:1 Node Protection

1:1 node protection has similar behavior as 1:1 link protection, it also important to maintain that node protection can recover from link and node failure, as shown in the figure (3.4), R2 as the PLR in case of R3 fails, will create 3 detour LSPs, one for LSP-A , the second one for LSP-B and the last one for LSB-C. PLR still uses the same rule as in 1:1 link protection, PLR consult the TED to find the best alternative path for each LSP's tail-end. Thus, LSP-A and LSP-B MP will be the tail-end. However, MP for LSP-C is R8.

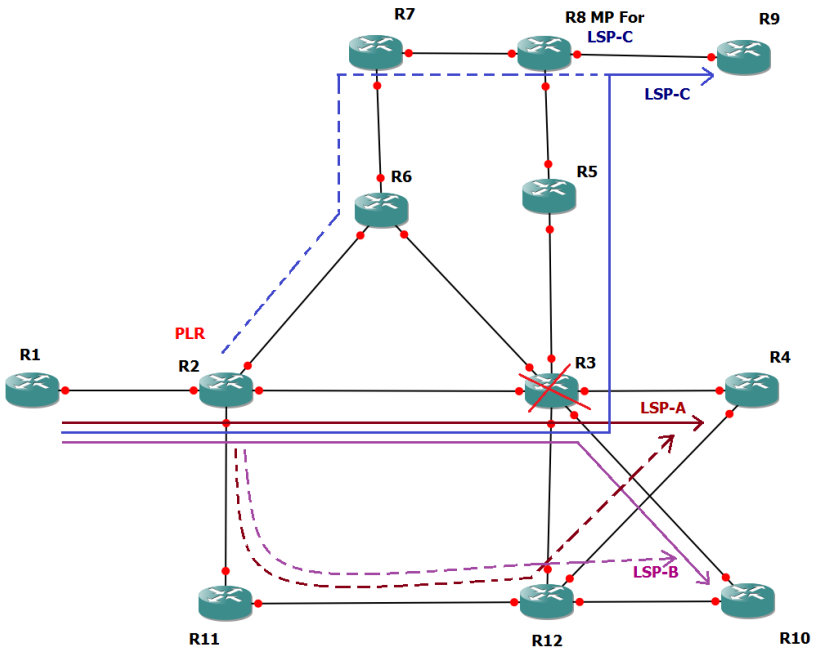


Figure 3.4: MPLS FRR 1:1 Node Protection

It should be clear that with 1:1 protection each LSP will have dedicated detour even if multiple detours share the same PLR and MP.

3.4.4 N:1 Node Facility Protection

This type of protection is also similar to link facility protection, the difference if that with link facility protection the PLR try to find the best path to the LSP next hop, in node protection since the next node is not working, node protection tries to find the best alternative path to the LSP next-next hop.

As shown in the next topology, LSP-A, and LSP-B next-next hop is the same IP address, which is R13, thus, both LSA-A and LSP-B will share one protection tunnel. However, LSP-C next-next hop is R5, thus, LSB-C protection tunnel ends up taken suboptimal path, because the traffic must go to R5 first then from R5 to the tail-end.

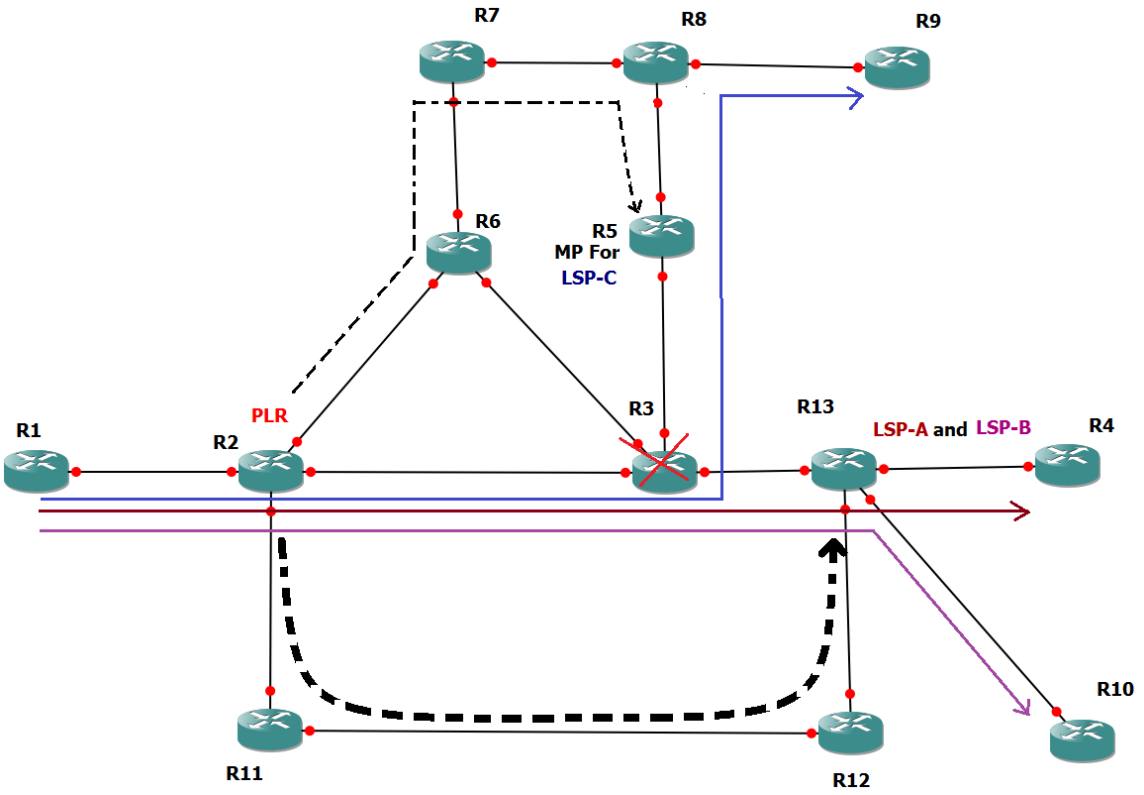


Figure 3.5: MPLS FRR 1:1 Node Protection

3.5 Protection Paths Computation, Signaling and Traffic Forwarding

The head-end router is responsible for announcing which LSP need to be protected; the head-end may have many LSPs, but not all of them may require protection. Thus, based on the LSPs requirements and the LSPs implementation plan, the head-end is configured to protect the desired LSPs.

Using RSVP path message, the head-end informs all nodes along the path that this LSP needs be protected. The head-end can also specify the backup path constraints, for example, the head-end can set the bandwidth reservation requirements, the number of hops allowed, and setup/hold priorities for the backup path. This information is signaled from the head-end to all PLRs along the LSP path by using fast reroute object message. However, in 1:1 protection some parameters such as bandwidth, link colors constraint can be inherited from the main LSP.

The next two sections discuss and analyze in details fast reroute mechanisms for path computation, LSP signaling and the operation of the labels when the traffic is taken the protection path.

3.5.1 Path computation for Link Protection

There are two methods to Forwarding the traffic over the backup path from the PLR to join the LSP at MP. First, let us analyze the forwarding process when facility protection is implemented. In this case, the rule to merge the traffic back to the main LSP is that the traffic must use the same label as it would be before the failure. To achieve this rule, PLR must push the backup path label on top of the main LSP label, and one hop before meeting the MP, PHP must be performed to remove the backup label and send only the main LSP label. The next example illustrates the process of facility protection label binding and forwarding.

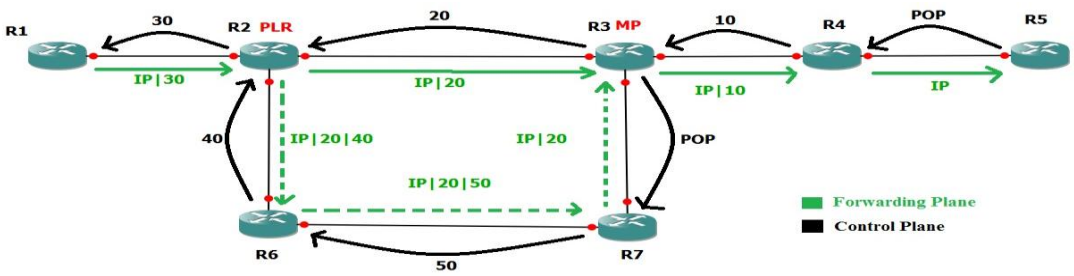


Figure 3.6: Link Facility Protection Path Computation

As shown above, the main LSP label binding is synchronized from the tail-end R5 to the head-end R1, each router informs the upstream neighbor about the label it must use for each LSP. The backup path will also be synchronized the same way from the MP R3 to the PLR R2.

When a packet is being forwarding in the main LSP, R1 will push label number 30, R2 swaps 30 label with 20, then R3 also swaps 20 label to 10. R4 pops the label 20 and sends the packet to tail-end R5. However, when the link between R2 and R3 fails, R1 push label number 30. R2 the PLR swaps the label 30 with 20 and pushes a second label 40, then it forwards the packet to R6. R6 swaps the outer label from 40 to 50. R7 pops the outer label 50 and sends the packet with inner label 20 to the MP.

The idea is simple in order for the MP join the backup LSP to the main LSP, MP must receive the same label as before the failure because at the MP there is no new forwarding table. In addition, the number of LSPs in facility protection does not affect the forwarding table at PLR, because facility protection uses in bypass tunnel for all LSPs. Furthermore, MP upstream neighbor must perform PHP.

Second, let us discuss the control plane and forwarding plane in one-to-one protection is. Since the number of detours with 1:1 protection is equal to the number of LSPs passing the link. Thus, using the same method as in facility protection is not sufficient. Instead, in 1:1 protection the MP must have forwarding information about each detour. The next figure simplifies the process.

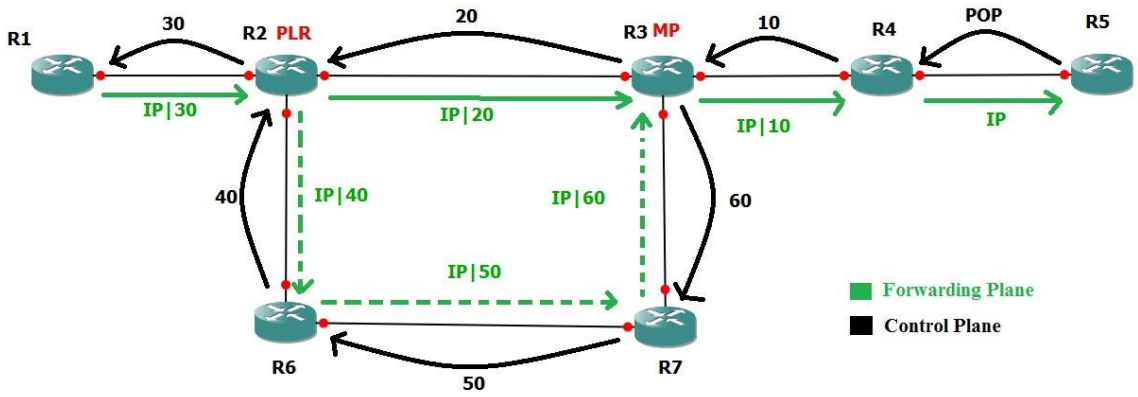


Figure 3.7: Link 1:1 Protection Path Computation

As shown above, in the case of failure in the link between R2 and R3, PLR and MP must have separate forwarding states for each LSP. It is clear that with 1:1 there is no label stacking, as only one label is used.

3.5.2 Path computation for Node Protection

Node protection sets up bypass tunnel to the next-next hop of LSP path in facility protection and to the tail-end for 1:1 protection. Therefore, 1:1 node protection has the same process as in 1:1 link protection.

However, in N:1 node protection the procedure is different, the PLR must swap the ingress label and push a second label, this label is what the next-next hop expect to receive in order to rejoin the traffic arriving from the backup path to the main LSP. The next figure illustrates the process.

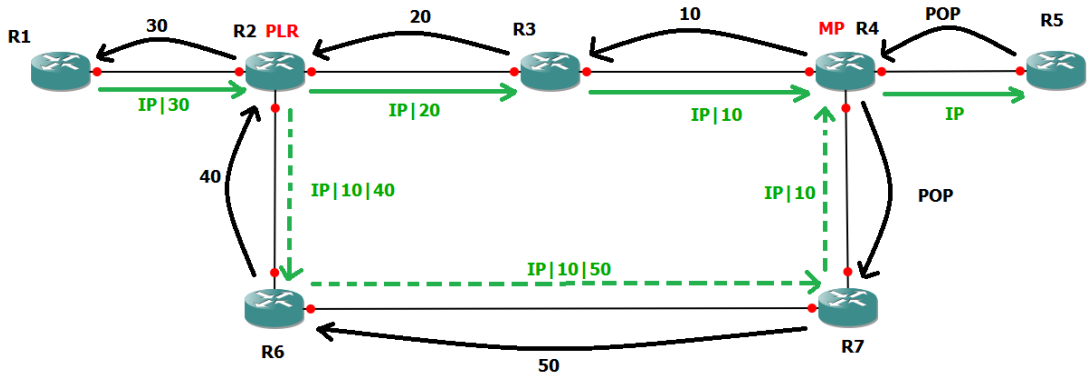


Figure 3.8: Node Protection Path Computation

As shown above, after PLR detects the failure, it switchover the traffic to the backup path. In details, R2 the PLR removes label 30 and pushes two labels the inner label 10 which is what R4 the MP expects and the outer label 40 for the backup path. R6 swaps the outer label from 40 to 60, and then it forward the packet to R7. R7 pops the outer label 50 and forward the packet with only 10 label. Once the packet arrives at the MP, the MP maps the traffic to the main LSP. In this case, similarly to link facility protection, the MP does not have new forwarding status in order to join the traffic coming from backup path to the main LSP, only PLR needs to prepare new forwarding status with labels that MP expects.

For further analyses, one question rises, how the PLR can know the MP label and IP address. The PLR finds the IP address of the next-next hop recorded in RRO, then it uses the IP as loose hop to

signal the path. However, by default, the RRO message does not carry any label information for the upstream nodes. Thus, it is very important to define new flag for label recording in the session attribute object.

4.0 CHAPTER FOUR

4.1 Introduction

So far, we have discussed and analyzed MPLS-TE LSPs being established, maintained and protected inside a single domain. Intra-area/level MPLS-TE benefits from the fact that all nodes share the same database. Thus, end-to-end LSPs with LSP constraints can be easily established, maintained and protected.

However, the global internet consists of multiple IGP domains and autonomous systems. In details, there are many reasons for dividing the network into multiple domains and ASs. The following points explain the reasons behind that:

- Scalability issues: running single IGP in large networks, results in scalability issues, for instance with single link state domain implementations, one single failure in the network results in LSAs flooding and SPF recalculation. In addition, as the network prefixes/nodes increase each node must maintain a larger database. It is known that as the network grows larger the potential of failures increases as well. Thus, the network may end up dropping packets as the CPU/RAM resources are over-utilized due to SPF calculations and the increase link state database. Thus, in order to have scalable and resilience network, it is necessary to divide the network to smaller sub-networks.
- Business-driven: to allow competition between service providers, the internet must be split into multiple autonomous systems, where each AS hides its internal infrastructure from other competitors. In addition, having one large network makes network-managing extreme difficult. Instead, dividing the network to smaller sub-network, where each group can administer their part of the network.
- Platforms vendors Compatibility: it might be hard to run one IGP across different vendor's devices or with the same vendor but with different platforms. Thus, it makes sense to divide the network based on this condition.

Normal users do not have knowledge about the network obstacles network-engineering encounter, and they demand the same quality of services even if the services they need has to span multiple sub-networks. For example, a client demand L3-VPN between two sites in different domains, this L3-VPN will be used to carry real-time traffic. Thus, the clients want to

ensure that there is no traffic loss so whatever. Another example, a service provider has voice gateways in different domains, thus, high availability LSP must be established and protected across multiple domains.

In this case, MPLS-TE needs to be established, maintained and protected between multiple domains. Achieving that is somewhat difficult for the following reasons:

- In single domains with multiple area/level design: the issue that each area/level has its own link-state database. Thus, the head-end cannot use CSPF, therefore, cannot calculate the entire LSP end to end. Because there is end-to-end TED is not available. In fact, the head-end can only calculate the path to the Area Edge Point (AEP. In total, there are two methods to signal LSP across multiple domains:

    1. Each APE must inject summary routes about other area/level, this way the head-end can find out which AEP is better to signal the LSP.
    2. The nodes inside the network do not need to know the overview to other area/level in the network. In this method, when an LSP need to be established the head-end consult a remote device which has an overview of the entire network.

- In case the LSP crossing multiple ASs, the same path setup, path computation and protection in inter-domains can be used, if the administrative domains can reach an agreement, the following points are the concerns that must be addressed:
    1. Record Route Object (RRO) has all IP addresses for the LSP downstream node; these internal addresses can be used to attack the downstream AS. Thus, before sending RRO message to the upstream AS, the RRO message must be filtered or modified.
    2. The number of LSPs setup and re-optimization requests must be agreed upon between the two parties. LSPs setup and re-optimization can be used for deny-of-service attack. In addition, LSP setup and re-optimization request must be initiated from a valid peer. Thus, RSVP authentication must be used.
    3. LSPs constrains translation should be also included in the agreement.

4.2 Inter-Domain Path Setup

There are three methods used to for Inter-domain/As path setup, as discussed in the following points:

1. Contiguous LSP: In this method, an end-to-end LSP is established across multiple domains. In fact, LSP signaling is done per hop basis between adjacent LSRs. The next figure shows an example of Contiguous end-to-end LSP.

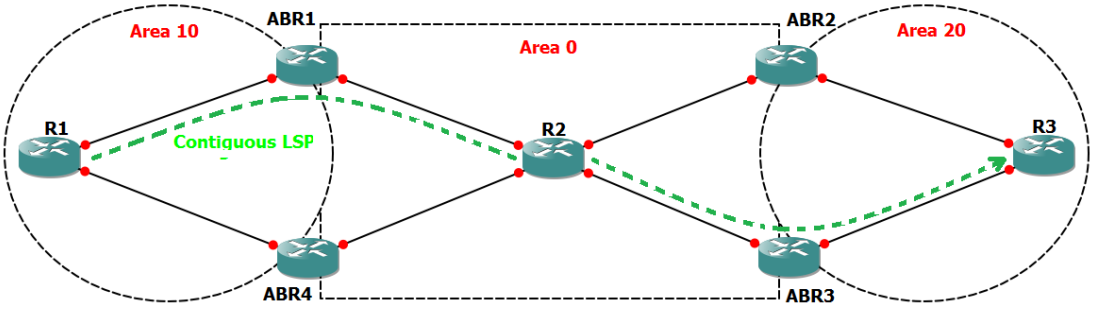


Figure 4.1: Contiguous LSP Path Setup

2. LSP stitching: In this case, the end-to-end LSP between the head-end and the tail-end consist of small LSPs where each of these LSPs is established within a single domain. Then these LSPs are stitched together at the border router. The next figure shows an example of stitching end-to-end LSP.

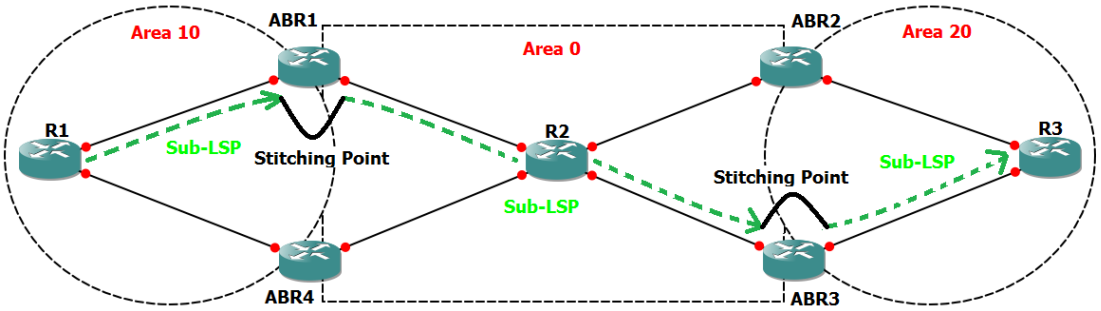


Figure 4.2: LSP stitching Path Setup

3. LSP nesting: In this technique, the end-to-end LSP are tunnels inside another LSP per domain. The next figure shows an example of nesting end-to-end LSP.

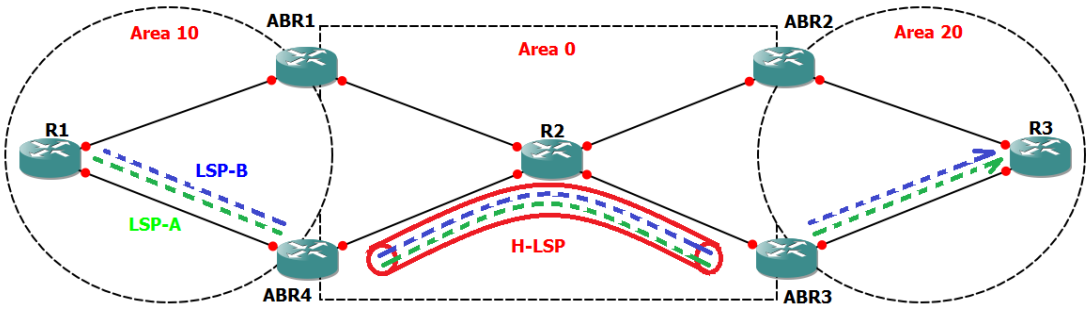


Figure 4.3: LSP nesting Path Setup

4.3 Inter-Domain Path Computation

Regardless of which signaling method is used, there are two ways to calculate the LSP path across multiple domains, it can be inter-domain path computation or per-domain path computation. In the case of Contiguous LSP setup, one might think that all LSRs along the path from the end-end to the tail-end across multiple domains can be signaled with the help of Explicit Route Object (ERO). In fact, the entire path calculation must have a multi-domain view, as results, an offline tool must be used. Think of offline tools as an external device that has all the requirements overall domain to establish Contiguous LSP.

It is also possible to do the calculation per domain as piece per piece basis because each domain knows the TED for the local domain and knows exit point such as ABR to other domain/destination. In addition, in this case, the head-end does not know the entire path. Instead, the head-end signal the LSP to the best exit point, then this exit point calculate the path to its best exit point, once it did the new path is added to ERO. This process is known as ERO expansion.

One major issue, with an LSP crossing multiple domains is that each domain may have different meaning and policies for LSP constraints, for example, different domains/ASs may use different class type for voice and different link color to high latency links. Therefore, domains/ASs administrations must agree on some policy and appropriately translate the constraints.

4.3.1 Per-Domain Path Computation Challenges

It is necessary to use per-domain path computation when the head-end does not have a full view of other domain. Therefore, the path computation must be performed per domain basis. In

fact, the head-end calculate the path to the domain edge point, then this domain edge point calculates the path to tail-end or to the another domain edge point that leads to tail-end. This way of path computation assumes that the reachability to the domain edge router is known.

The border router either can be configured at the head-end as loose hop or it can be discovered dynamically. After the path is computed any LSP setup can be used.

Per-domain path computation has one big disadvantage, which is the path is not always the optimal. In details, the path within each domain can be optimal, but the end-to-end LSP that uses the sub-path in each domain can be suboptimal. The next example illustrates this issue.

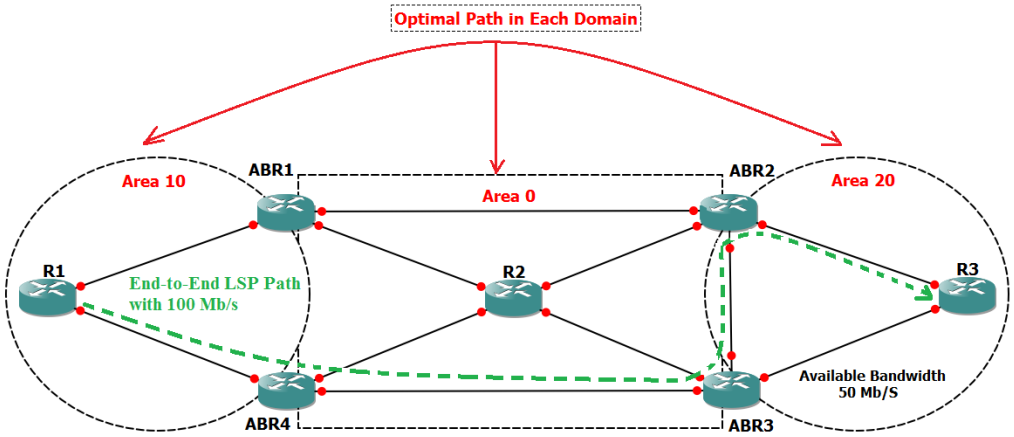


Figure 4.4: Non-Optimal End-to-End LSP Path

As shown in the figure, from each area point of view the LSP is taken the optimal path, but that is not totally true. If the Link between ABR3 to R3 can satisfy the LSP BW constraints then the path is optimal, but instead, when the LSP in area 20 is calculated, ABR3 finds the best path that satisfies the LSP constraints to R3 is via ABR2. As results, this end-to-end LSP path is suboptimal; it would be optimal if the end-to-end path were R1-ABR1-ABR2-R3.

Another interesting issue happens when the LSP setup fails because of old traffic engineering information, for example, after the path is calculated and the LSP setup starts, the resources that were available during path calculation is not available anymore because another LSP has taken the resources, this results in LSP setup failure. In a single domain, MPLS-TE this issue is solved by sending RSVP error message to the head-end, and since the head-end has the entire

view of traffic engineering information, the head-end recalculate another path meets the LSP requirements.

However, in multiple domain LSPs setup, sending the RSVP error message to the end-end does not solve the issue because the end-end might use the same path because the head-end does not know any TE information out of his domains. For further analysis of this issue let us consider the next scenario as shown in the figure (4.5).

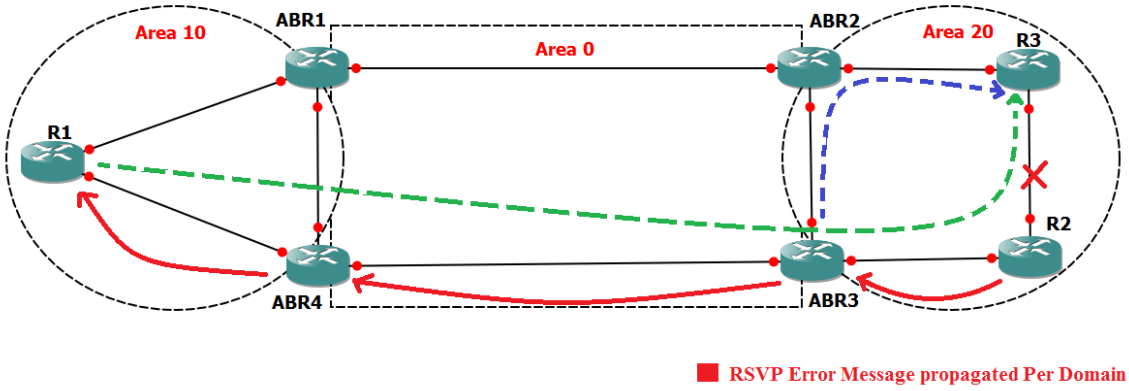


Figure 4.5: RSVP Error Message Propagation

In the above figure, an LSP need to be established between R1 and R3, after the path is computed in each area the end-to-end path takes the path R1-ABR4-ABR3-R2--R3. Once the setup started the link between R2-R3 fails to provide the resources because another LSP is already established and took the resources. As a result, end-to-end LSP setup failed.

Since sending RSVP error message does not solve this issue, an RSVP message must be sent to the first node that passes the LSP inside the domain that originates RSVP message. In this scenario, RSVP error message is sent to ABR3. Once the RSVP error message is received by ABR3, ABR3 recalculate new path, thus, ABR3 selects the path ABR2-R3.

If ABR-3 cannot find alternative path, it passes the RSVP error message upstream to the first node, in this scenario, ABR3 forward the message to ABR4, then ABR4 takes responsibility to find alternative path, if ABR4 cannot find any paths meet the requirements, the ABR4 forward the message to the head-end R1.

This process of trying to solve the issue locally within the domain before reported to the first upstream node is known as crankback. This method works great with nested or stitched path setup because the LSP is truly not end-to-end. In fact, with these techniques, end-to-end LSP is made of partial LSP within each domain.

To sum up, using crankback technique is very powerful to find an alternative path close to the point of admission control failure. However, finds the alternative path that meets the LSP constraints takes time, and the issue with the non-optimal path can still exist.

4.3.2 Path Computation Element (PCE)

Intra- domain MPLS-TE is efficiently capable of establishing, maintaining and protect end-to-end LSP. However, Inter-Domain MPLS-TE at this time is imperfect when it comes to path computation, the reason for that is inter-domain MPLS-TE database has limited TED view, and it cannot synchronize the TED for multiple domains. In fact, the head-ends TED is limited to the local domain where it belongs. In addition, Some constraints are only known by the head-end all other nodes are not aware of these constraints such as link colors, thus, link color using normal per-domain path calculation cannot be used across multiple domains. Thus, it is impossible for the head-end to use CSPF to compute an end-to-end path to the tail-end.

Having reliable LSP that crossing multi-domain is desired for many reasons, for example, voice traffic passing two or more domain, in this case, the voice gateways may be in different domains, or they might be in the same domain, but they need to establish an LSP over the neighbor domain. For example, a redundant path must do via a second domain, or they might not have the capacity within the local domain.

As discussed in the previous section, computing the path per domain basis is not efficient. Thus, it is very important to find a solution that provides an optimal, constraints guarantee end-to-end path, in addition, to provide protection against domain edge router failure. Therefore, a new mechanism in needed to provide special path computation and cooperation between nodes different domains.

Path Computation Element (PCE) is an object designed to be capable of computing constraints LSP path from the head-end to the tail-end. The PCE entity is an application that can be located

within a network node such as LSR or on an out-of-network such as a server. PCE provided a solution to compute the path within a single domain or multiple domains. The following points include PCE implementation methods:

- Single PCE path computation: In this case, single PCE element is used to compute a path within a domain. It is possible to have multiple PCEs in one domain, but only one can perform path computation.
- Multiple PCEs path computations: in this implementation, Multiple PCEs are used to compute the path.
- Centralized computation model: in this method, single centralized PCE is responsible to path computation for all nodes in one domain or multiple.
- Distributed computation model: in this model, multiple PCEs as responsible for path computation inside one domain with the cooperating among PCEs.

For inter-domain MPLS-TE path can be computed used only two models, either Distributed computation model where one or more PCEs are responsible for each domain or centralized model where only one PCE is responsible for path computation over multiple domains. Placing PCE in the LSR might results in some issues, as they are listed in the next points:

- Path calculation with large numbers of LSPs may consume CPU resources. Thus, out-of-network PCE solution may be required.
- Limited TED visibility: in the case of the head-end and the tail-end in a different domain, the head-end can only see the topology up to the domain border router. This issue can be solved by using loose hops to establish an LSP. However, the end-to-end path may end up taken non-optimal path. Thus, PCE solution can be used in this case, for example, distributed computation model, where each domain PCEs is responsible for its own domain part, but PCEs in different domains are cooperating with each other. A better solution is to use centralized computation model, where one PCE computes the path end-to-end, however, this method has scalability issue because it needs to have full database view for all domains to be located in one PCE.

4.3.2.1 LSR with CPE functionality

In this case, as shown in the below figure (4.6), a router is functioning as CPE. A link-state routing protocol is still used to exchange TED. This TED is severed to the PCE, and once the LSP

need to be established, the signaling engine makes a request to the local PCE. The PCE does not use CSFP, it uses a more advanced algorithm to compute the path using the TED, and then PCE sends the computed path back to the signaling engine to signal the path starting from its neighbor.

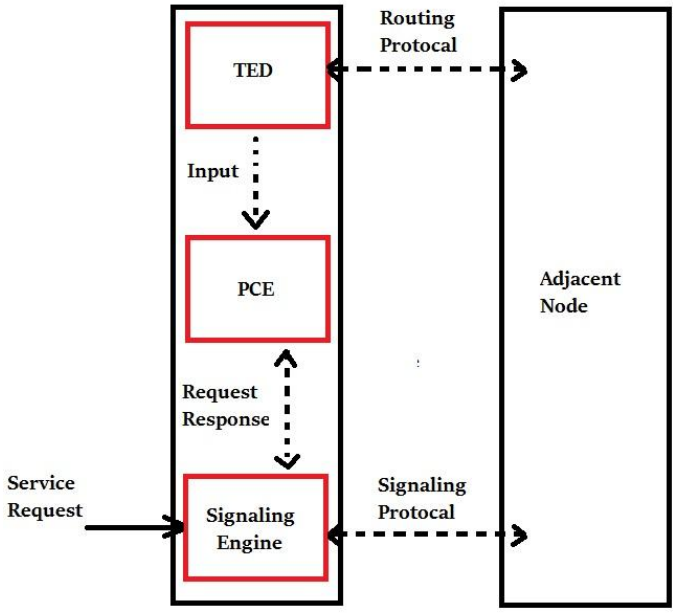


Figure 4.6: LSR with CPE functionality

4.3.2.2 External CPE functionality

This case is similar to the previous one expects that CPE is an external device such as a server. The CPE need to maintain synchronized TED, in details, two methods can be used for PCE to acquire TED, first, sniffing link state advertisements or by sending a request to the PCC.

Path computation client is a term used to describe the LSRs that need to establish an LSP or it can be any LSR in the network. The communication between PCC and PCE uses PCEP. In fact, PCPE uses RSVP objects to carry and exchange information that is needed LSP path computation.

PCCs can locate PCE location either statically or dynamically. In a static method, the IP address of PCE is statically configured in PCC. However, this static implementation is not recommended for the next two reasons, first, in a very large network, it has to manage a large number of PCCs and PCEs, second, statically defined PCEs does not have the ability to switch to another PCE in

the case of failure. Thus, for easy management and high availability dynamically PCE discovery is very recommended. The help of link state extensions as in RSVP auto-meshed solution does this auto discovery technique.

The operation as shown in the below figure, the head-end sends a quire the external PCE to request path computation , the PCE runs its algorithm and sends back the results to the head-end.

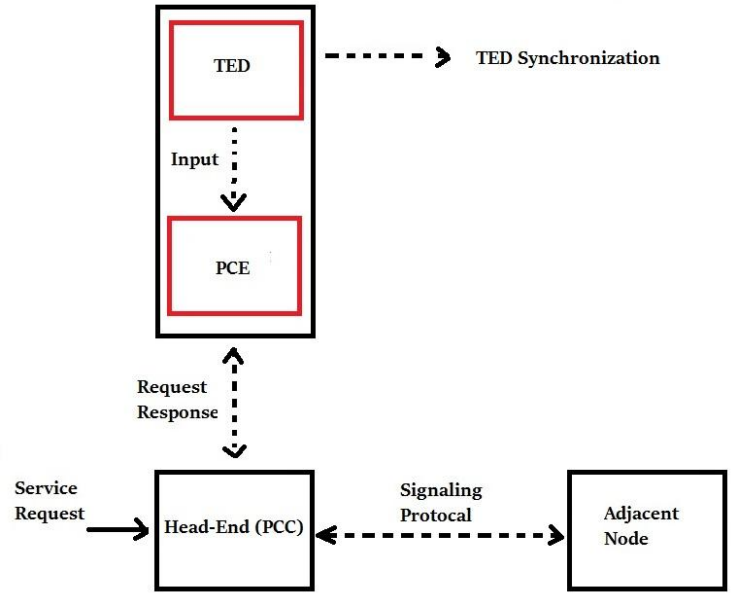


Figure 4.7: External CPE functionality

4.2.2.3 Multiple External PCEs Path Computation

More than one PCE can be used to calculate end-to-end LSP. In fact, two methods can be used, first, multiple independent external PCEs , in this case, each PCE is responsible for computing the path to the best domain edge router, then the domain edge router requests a path computation from the second PCE, then process is repeated per domain, where each domain has its own PCE. The second method is multiple cooperated external PCEs, in this case, there is inter-PCE communication among PCEs. In fact, each PCE send a request to other PEC to find the end-to-end best path. PCEs are communicating with each other by using PCE communication

protocol (PCEP). The function of this protocol is to exchange TED-related information in each domain.

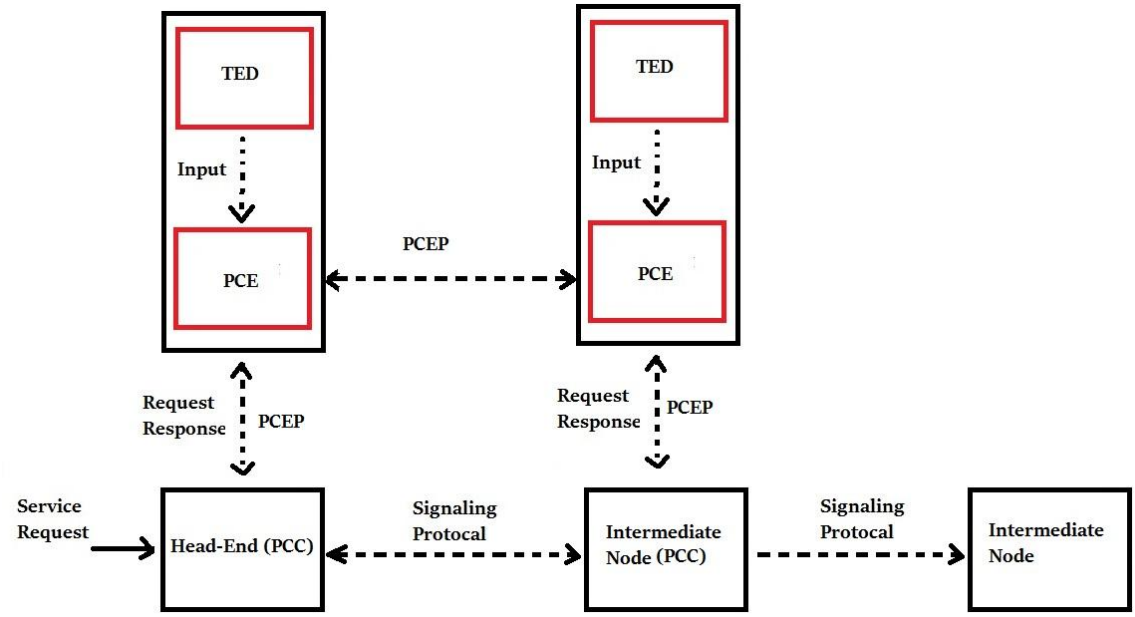


Figure 4.8: Multiple External PCEs Path Computation

4.3.2.4 Centralized PCE Path Computation

In this method, One PCE or more have a complete view over all topologies in all domains, thus, PCC sends a path request to one of PCE, and this PCE calculate the end-to-end path and give the sends the result to the PCC. It is highly recommended that more than one PCE is available to PCC request in case of PCE failure or PCE overloaded.

4.4 LSP Re-Optimization

LSP, as discussed in previous sections, might end up using suboptimal path. Re-optimization is a process of finding a more optimized path and switchover the traffic from old non-optimized LSP to new more optimal LSP. The switchover follows the make-before-beak rule. Therefore, the new LSP have to be established before switchover, and after the switchover is done the old LSP is torn down.

As mentioned in previous sections, in a single domain, the head-end is responsible for re-optimizing and signaling the entire path. In multiple domain seniors, however, both path computation method and LSP signaling influence the re-optimization process.

In the case of contiguous LSP setup re-optimization then the head-end must be involved in the process. On the other hand, per domain LSP path computation with stitching or nesting setup method, the re-optimization is done per domain without interfering from other domain or involving the head-end. This separating in the re-optimization process helps the overall scalability because out of domain LSR are not affected. In addition, because of make-before-break is local to the domain, other domains are not affected by the temporary duplication of the resources reservation.

4.5 Inter-Domain LSP Protection

Most protection methods for Intra-domain LSP protection can be used with Inter-domain LSPs. However, there is some point must be addressed. In the case of having two end-to-end paths, one is primary and the second is secondary. It is mandatory that both LSPs do not share any link to provide full protection. As we have seen this rule can be achieved in the intra-domain case by using strict hops along the path. However, in the case of inter-domain, this is not possible because the head-end has limited visibility about other domains. One way to eliminate this issue is by using PCE centralized method, where the PCE has knowledge about the entire domain.

In the case of inter-domain, if both PLR and MP in the same domain, the same protection mechanism for local protection (Node and Link) can be used. However, if PLR and MP in different domains, there are two issues, first, how the backup path is calculated? Second, how the MP is identified? To explain the two issues let us consider the following scenario.

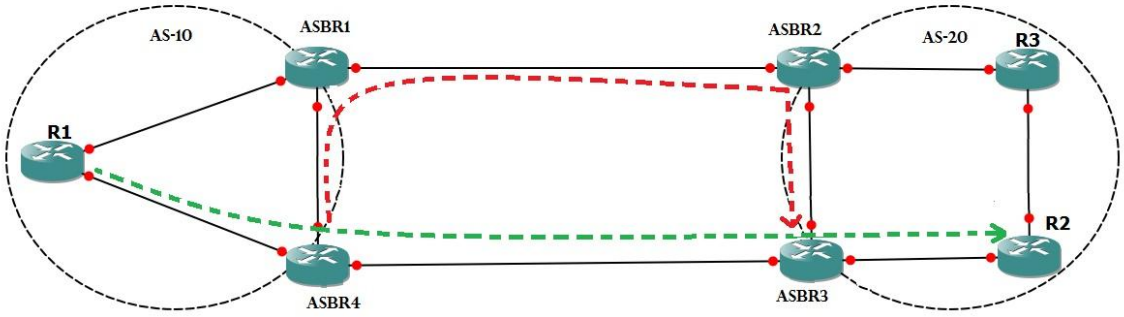


Figure 4.9: Inter-Domain LSP Protection

As shown above, in the diagram an inter-domain LSP is established from the head-end R1 to the tail-end R2. In addition, a local protection is needed between the two domains. First, the link between ASBR4 and ASBR3 needs to be protected, thus, a backup detour ASBR4-ASBR1-ASBR2-ASBR3 must be established. Doing this raises two issues, first, how to locate the MP, in intra-domain cases the MP is located by checking RRO message to find MP id. In inter-domain the address of the other domain are not advertised in the link state, thus, to locate MP the node ID (loopback address) must be advertised in IGP.

After locating the MP, the second issue is finding and computing a backup path to MP without using the primary path. The PLR does not have TED about the other domain. Thus, one of Inter-domain path computation must be used.

In the case of node protection, the issue is more complicated. In a single domain, MP address is the next-next hop. However, in inter-domain, the MP location depends on the LSP setup method. If the LSP setup is contiguous then MP is the next-next hop, and if the LSP setup uses stitching or nesting methods then the MP can be only tail-end of same LSP segment. As results, for this long protection to the from ingress domain edge router to the egress domain router, the reservation will be made throughout all links.

5.0 CHAPTER FIVE

5.1 Introduction

Virtual Private Network (VPN), is a technique used to provide a secure logical connection between two or more private sites over a public network security. These VPNs can be a point to point (P2P) or point to multipoint (P2MP).

There are two methods a VPN can be characterized under; it can use Overlay VPN module or Peer VPN module. In Overlay VPN module, a layer 2 P2P connection over ATM, Frame Relay can be used to provide connectivity between two customer sites. In this case, the traffic between passing the public network is encapsulated over L2 tunnel. In addition, a layer 3 P2P or P2PM VPNs can also be used via IP tunnels such as Generic Route Encapsulation (GRE) or IP Security (IPSec). In overlay case, the public network does not participate in any kind of routing with a customer site, instead, it provides a logical tunnel to overlaid the traffic on top of the provider's infrastructure and the customer must take care of the routing parting between its own sites. One major diamante of this module is the use of bandwidth inside the public network, for instance, in L2 overlay VPN, once the circuit is established and the bandwidth is located, the public network cannot reuse this capacity even if the customer traffic is idle.

In peer VPN module, the customer sites do not peer with each other directly. In this module, the public network the customer edge router (CE) peer with directly connected public network edge router (PE). Thus, the service provider will route the customer traffic, and the customer does not participate in traffic routing anymore. However, this method raises extra overhead on the provider network. For example, because know the PE must maintain all customers routes, and the PE might be interconnected to multiple customers, the PE must maintain forwarding table for each customer and it must stop route leaking between customers. In addition, the PE must distingue each customers routes if there is overlapping between private IP blocks that each customer use. Multiple techniques have been used to solve these issues such as using Access-lists to eliminate route leaking between CEs and per customers PE solution. Currently, Multiprotocol-Border Gateway Protocol (MP-BGP) is the being used to provide L3-VPN. This chapter illustrates and analyzes MP-BGP L3-VPN within a single domain and multiple domains.

5.2 MP-BGP L3-VPN Solution

MLPS L3-VPN solution uses MP-BGP routing protocol and MPLS LSP to provide tunnel based VPN solution. The goal is to achieve a scalable method that provides total traffic isolation between different VPNs even if the IP blocks are overlapped. In fact, MP-BGP L3-VPN works by using BGP to provide connectivity over the public network between PEs; Second, BGP is also used to advertise CEs route information over the public network in a scalable manner. MPLS is used to tunnel the CEs traffic from one PE to another PE. As described in chapter 4, MPLS provider or RSVP-based LSP tunneling capabilities. Therefore, efficiency and protection are guaranteed. The following sections illustrate in details how the overall MP-BGP L3-VPN solution works.

5.2.1 Virtual Routing and Forwarding (VRF)

Virtual routing and forwarding (VRF) are a technique that creates multiple instances of a routing table to co-exist in the same router where each instance represents one customer forwarding table. These routing instances are totally independent, therefore, the same or overlapping private IP blocks can be used by the customers without conflicting with each other, as shown in the next figure(5.1).

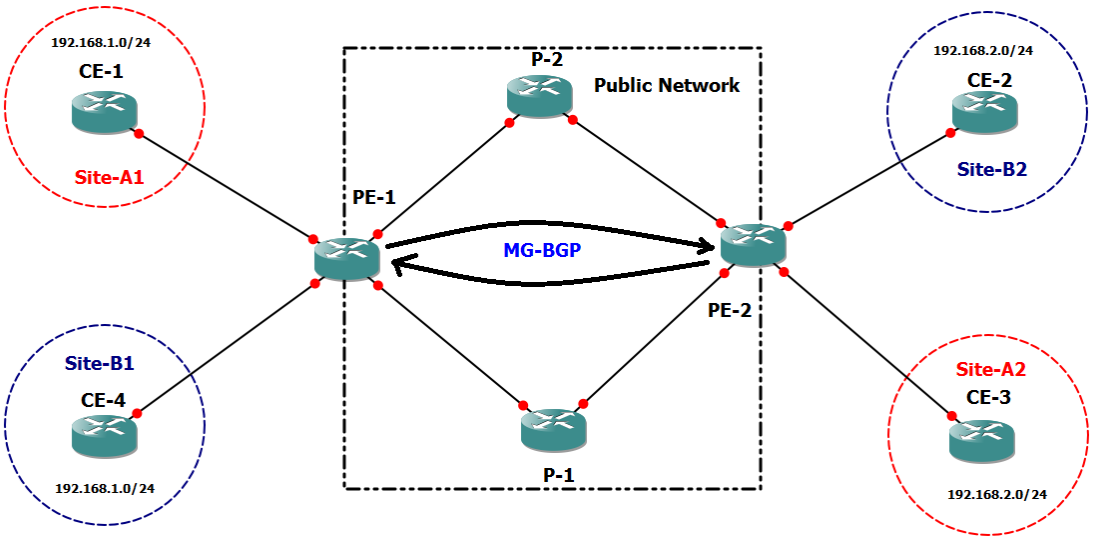


Figure 5.1: Virtual Routing and Forwarding

As shown in above two customers each has two sites, customer A needs an L3-VPN between site A1 and A2 and customer B needs an L3-VPN between site B1 and B2. Both customers use

the same IP address plan, site A1 and site B1 uses the same IP subnet 192.168.1.0/24 and there are connected to the same PE. Since VRF can solve the overlapping issue, each site requires unique virtual forwarding instance to avoid the overlapping. Thus, PE-1 and PE-2 will end up having two VRFs. Each VRF has its own forwarding table for its mapped customer. At this point, each PE can eliminate the overlapping, but how the other PE will distinguish between each customer routes? The next section illustrates how the routes are differentiated within the public network.

5.2.2 The Route Distinguisher (RD)

Route Distinguisher is used to distinguish customer's private routes. In fact, Before advertising a customer VPN route by using MB-BGP, the PE router attaches an RD to each route, each customer must have unique RD to avoid VPN routing from overlapping. After the routes are received by MP-BGP peer the RD is removed and the new VPN routes are placed in the appropriate VRF.

The route distinguisher is an eight octet's field. Thus, the total VPN route length is 12 octets. RD does not help identify the VPN, the RD only ensure uniqueness of VPN routes carried in MP-BGP. Therefore, using one RD per VPN per PE works perfects. The process of sharing VPN routes in explained in the next section.

5.2.3 The route target (RT)

Route targets are used to share VPNs routes among different sites. This route target uses BGP extended communities before the PE advertises the VPN route is adds the route target to the route by using BGP extended communities. Thus, once the another PEs receives the VPN routes they can decide which routes must go to which VRF/VPN table, and which route are discarded.

Route target is a 32 bits string with normal BGP communities. The first 16 bits reflects the AS number and the last 16 bits is a locally assigned number. However, with extended BGP communities, the locally assigned number is 32 bits.

The PE can attach one or more route targets to VPN route, the number of route targets depends on how many VPNs shared the same route. In addition, route targets can be configured as

export and import. Export means, attach this community to this VRF route before the PE distribute it to other PEs, Import means, that each received a route that has the same export route target as my import route target will be important to the VRF that own the import route target.

Manipulating RT can results in many kinds of designs between the same sites, for example, route target can be used to have full mesh between all sites. In this case, each site can communicate directly with all other sites. In fact, each site must have the same import and export route targets.

An alternative design is a hop and spoke, in this case, all site are communicating with only one site directly known as hop, all other sites known as spokes uses this hop as transited to reach other spokes. This design is highly recommended in case if the traffic must be inspected by a central firewall.

Another design is known as Overlapping VPNs, this design is been used when there is services such as database or voice gateways need to access by different VPNs. Thus, each VPN that need to access the centralized services must import the centralized services export RT.

5.3 L3-VPN Control and Forwarding Planes

As discussed previously the VPN routes are advertised between PEs using BGP. Therefore, the PEs will use the IP address of PE whose advertises the VPN route as next hop. This technique raises an issue because the intermediate nodes (Core routers) between PEs do not have knowledge about the VPNs routes. Thus, tunneling must be used between PEs to prevent traffic dropped.

An MPLS LSP can solve this issue; each PE configured to has full meshed LSPs to all other PEs. These LSPs can be created using LDP or RSVP or LDPoRSVP based on the requirements and core design. Since it is tunnel based, the core routers only need to swap the outer label and forward the packet to the next hop.

Each VPNs will be associated with an MPLS label. In fact, once the VPN customer is defined at the PE. This PE bind a label for its VPN routes, and then it advertise these labels to remote PEs by using MB-BGP.

After the control plane is synchronized, the traffic can be forwarded, the forwarding process for the figure (5.1) is described in the next points:

- Site A1 needs to communicate with Site A2, both CE-1 and CE-2 has routes reachability to each other.
- CE-1 sends IP packets to PE-1, PE-1 checks the VRF forwarding table for CE-1 to locate the VPN label, and then it checks its global forwarding table to locate the transport label for PE-2 as the next hop.
- Then PE-1 push the two label the inner label to identify the VPN at PE-2 and the outer label to transport the packets over the core.
- Either P-1 or P-2 will be the intermediate node for based on the LSP path computation if PHP is not enabled on the network P-1 or P-2 will swap the outer label to another label based on MPLS binding then the packets are forwarded packets to PE-2. PE-2 will pop the first label, then it looks to the inner label to identify the VPN, after VPN is identified, the label is removed and pure IP packets are forwarded to CE-3.
- If PHP is enabled then P-1 or P-2 pops the outer label and forward the packets with VPN label only. PE-2 checks the label to identify the VPN, then it pops the label and forwards pure IP packets to CE-3.

5.4 Inter-AS L3-VPNs

So far, we have discussed Intra-AS MPLS L3-VPN. Inter-AS -L3-VPN is needed when a customer wants to establish L3-VPN between two or more sites that reside in different AS. Therefore, all ASs involving on L3-VPN must corporate with each other. Since MP-IBGP peering is not allowed between ASs. MG-EBGP must be used. In fact, there are three techniques to implements Inter-AS L3-VPN as listed and discussed below:

1. Inter-AS L3-VPN Option A.
2. Inter-AS L3-VPN Option B.
3. Inter-AS L3-VPN Option C.

5.4.1 Inter-AS L3-VPN Option A

Option A is the simplest option, this option treats each VPN as a customer by using back-to-back VRF, thus, each AS send a packet to its upstream AS as if it is an L3-VPN customer. In the example, as shown in figure (5.2), two sites in different AS needs to establish an L3-VPN connection, site A in AS 10 and site B in AS 20. In addition, any CE-PE routing protocol can be used between ASs.

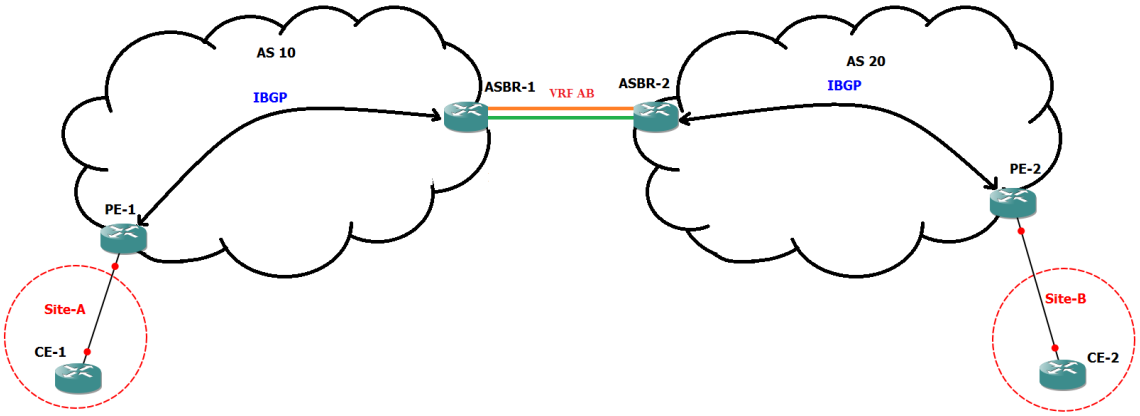


Figure 5.2: Inter-AS L3-VPN Option A

ASBR must use logical interfaces to support multiple Inter-AS VPN customers. ASBRs physical interconnection can be divided to multiple sub-interface, where each sub-interface represent one VPN customer, or it can use switch virtual interfaces (SVI), where the physical interface is set as L2 trunk and in each ASBR SIVs represents customers VRFs.

Let us assume that site-A want to communicate with site-B, in this case, CE-1 send native IP packets to PE-1. PE-1 pushes two labels, once label for the VRF and second outer label for the transport over AS-10, once the packet reaches ASBR-1, the label are gone and native IP packets are sent to ASBR-2. ASBR-2 pushes two labels, once inner label for the VRF and outer label to transport the packets over AS-20. When the packet arrives at PE-2, PE-2 checks the VRF label and forward native IP packet to CE-2.

It is important to notice that MPLS are not enabled between, and separate LSP is being used inside each AS. Therefore, MPLS-FRR is not possible in case of one of ASBRs fails and redundant ASBR is available. It is also important to notice that this method is not scalable, because, with the increase of VRFs numbers, the overhead is also increased at ABSR.

5.4.2 Inter-AS L3-VPN Option B

Option B uses only one EBGP session for all VPNs. Therefore, the scalability improved comparing to the Option A solutions. In option B there is IBGP peering inside each AS between ASBR and PE, and one EBGP peering between ASBRs. It is also important to mention that an LSP across multiple AS is needed, an end-to-end LSP between PEs is a necessity in order to forward the traffic between VPNs sites.

This end-to-end transport LSP depended on two things: First, if the PE next hop is the local ASBR, then shown in the example, three LSP is required; one LSP from PE-2 to ASBR-2, a second LSP is between ASBR-1 and 2, and third LSP from ASBR-1 to PE-1. However, if the next hop is the remote ASBR, then only two LSP is needed, but the link between ASBRs must be advertised inside each AS. For instance, PE-1 will have one LSP from PE-2 to ASBR-1, then a second LSP from ASBR-1 to PE-1.

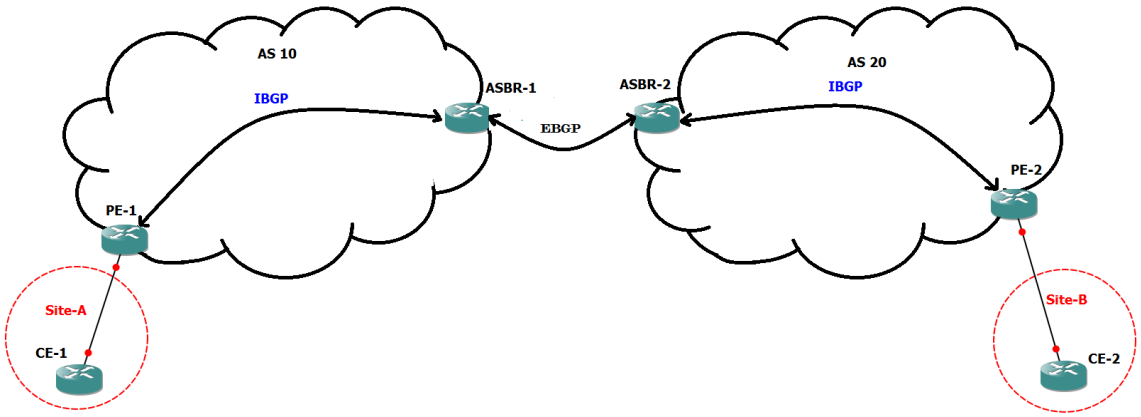


Figure 5.3: Inter-AS L3-VPN Option B

The end-to-end control plan for option B is described in the following points, assuming that the transport LSP next hop in local:

- PE-1 advertise VPN-V4 label to ASBR-1 for example X-1 label
- ASBR-1 creates a new label X-2 and advertises this new label to ASBR-2. In addition, ASBR-1 built a forwarding label to swap X-2 with X-1.
- After ASBR-2 receives X-2 label, it creates new label X-3 and forwarded to PE-2.In addition, ASBR built a forwarding table to swap label X-3 with X-2.

- Once the label X-3 received by PE-2, PE-2 install this VPN label in the forwarding table for that VRF.

Let us see the forwarding plane, once the PE-2 receives an IP packet for site-A, PE-2 push two labels once label for the transport and the inner label for the VPN which is X-3 in this example. This MPLS label will be swapped by ASBR-2 from X-3 to X-2. Then once ASBR-1 receives the MPLS packet is swaps it again by removing X-2 label and inserting X-1 label. Finally, once the packet received by PE-1, PE-1 checks the VRF label and forward native IP packet to CE-1.

Even though option B provide better scalability comparing to option A, the ASBRs still needs to maintain and advertised all VPN routes and their associated forwarding tables.

5.4.3 Inter-AS L3-VPN Option C

This option C is the best solutions for Inter-As L3-VPN because it removed the overhead on the ASBRs. As described in the previous section, option B has scalability issues because ASBRs must maintain the forwarding state for each VPNs. Option C eliminate the need to maintain the forwarding table by establishing multi-hop EBGP peering between PEs reside in different ASs, as we can see in the next figure (5.4)

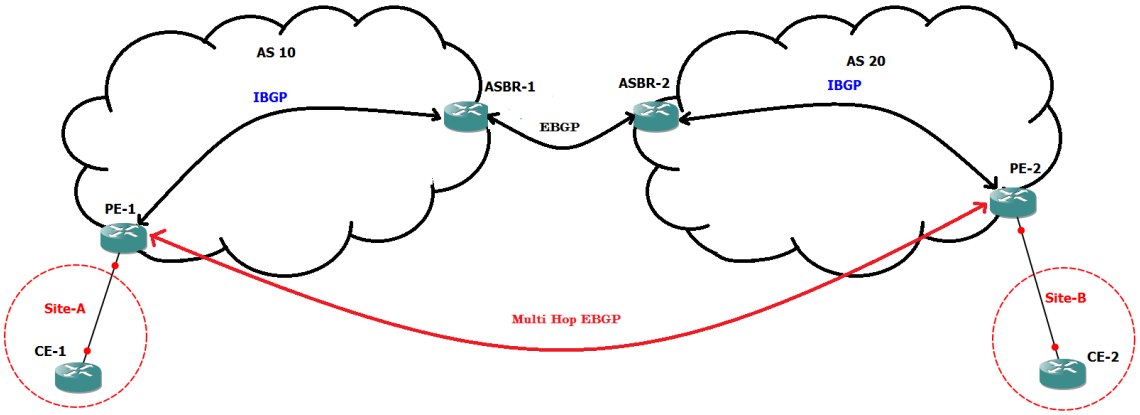


Figure 5.4: Inter-AS L3-VPN Option C

In order for this method to work, the PEs loopbacks addresses must be reachable from the two ends. This is needed because the next hop IP address for the VPN must stay the same in the entire path from PE to PE. However, due to security concerns and trust issues between ASs administrative, option C is undesirable and has very rare implementation.

This implementation, however, requires end-to-end LSP from one PE to the other remote PE. Therefore, any inter-domain LSP setup method can be used as described in chapter 4.

6.0 CHAPTER SIX

6.1 Introduction

In this chapter, Intra-Area, Inter-Area and Inter-AS Traffic Engineering are implemented and evaluated by using different singlings and path computations methods. The first section focuses on Intra-Area Traffic Engineering LSP implementation and protection. The second section, focus on Intre-area Traffic Engineering LSP implementation and protection. The final section, illustrate the Implementations of L3-VPN passing multiple ASs.

Due to hardware and software limitation, Inter-Area traffic engineering Implementations are performed on graphical network simulator using IOS-XR V 5.3.2. GNS3 is network simulator that uses real network operating system. GNS3 gives reliable results. However, the network scale and service emulation during the tests can be affected by the resources located to GNS3. In lab, Cisco 2921 platforms have been used for Intra-Area and L3VPN Inter-As traffic engineering.

6.2 Intra-Area Traffic Engineering

6.2.1 LAB Setup and Guidelines

- IP Plan: the subnet 192.168.0.0/16 is used in this lab setup for P2P connection between routers, whereas, the third octets represent the routers numbers sharing one P2P subnet and the fourth octet is used to represent the router number. This way of subnetting is very useful when the tests are running and at troubleshooting time. In addition, the router number in all octets represents each router loopback address.
- Topology: 6 P routers and 2 PE routers will be used, each PE is connected to a ubuntu virtual machine as shown in the figure (6.1)

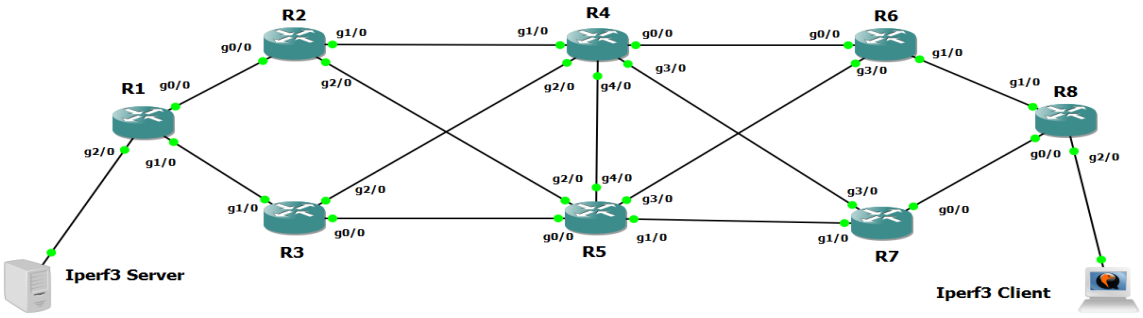


Figure 6.1: Lab Topology

- IGP Routing Protocol: OSPFv2 is used as the Interior gateway protocol.
- MPLS-TE is used to route the traffic over the network, different LSP setup methods will be used and multiple protection techniques will be tested per LSP setup method.

6.2.2 Intra-Area MPL-TE Implementation:

- L1: each interface is configured with an IP address as described previously. For example, the link between R2 and R4, R2 IP address is 192.168.24.2/24 and R4 IP address is 192.168.24.4/24, as for loopback address, R2 loopback address is 2.2.2.2/32 and R4 loopback address is 4.4.4.4/32.
- L2: All interfaces are 1G-Ethernet.
- L3: OSPFv2 is the IGP, the following points illustrate the configuration, the sample configuration shown below is taken from R1 configuration:
  - Enabling OSPF on the router:

```
#router ospf 1
 #router-id 1.1.1.1    # hard-code the OSPF router id
```

  - Enabling OSPF on interfaces: By default, Ethernet interface takes more time to form neighborship with other OSPF adjacency, because of the process to elect the DR and BDR routers. Thus, since only P2P Ethernet OSPF adjacency is needed, the interfaces are configured as P2P OSPF network. This method improves convergence time.

```
#interface GigabitEthernet0/0
 #ip ospf 1 area 0                # enable OSPF
 #ip ospf network point-to-point # for fast ospf adjacency formation.
```

  - Eliminate unnecessary OSPF messages: Any interface that needs to be advertised into OSPF and there is no OSPF adjacency via the same interface must be configured as a passive interface. Passive interface, eliminate OSPF adjacency and save resources (CPU and RAM).

```
#router ospf 1
 #passive-interface GigabitEthernet2/0
 #passive-interface loopback0
```

- MPLS-TE: since we are using MPLS-TE, RSVP must be enabled on each router's interfaces.

```
#mpls traffic-eng tunnels
 #interface GigabitEthernet0/0
 #mpls traffic-eng tunnels  #Enable TE-RSVP on the interface
 #ip rsvp bandwidth         #Assignee BW to be used by LSP. The default is 75% of the physical BW
!
#router ospf 1
 #mpls traffic-eng router-id Loopback0 # Source IP for MPLS-TE
 #mpls traffic-eng area 0              # OSPF area that shares TED information
```

- Configure MPLS-TE LSP tunnel: In this LAB an LSP tunnel is configured from R8 to R1, multiple LSP setups and path calculations are being used as described in the following points:

6.2.2.1 Contiguous LSP (Dynamic and Explicit path computation):

1. One LSP using CSPF:

```
#interface Tunnel81
 #ip unnumbered Loopback0                          # LSP source IP address
 #tunnel mode mpls traffic-eng                     # Tunnel mode is MPLS-TE
 #tunnel destination 1.1.1.1                       # LSP tail-end IP address
 #tunnel mpls traffic-eng autoroute announce       # install a route in routing table
 #tunnel mpls traffic-eng path-option 1 dynamic    # Path computation CSPF
```

2. One LSP using Explicit path:

```
#interface Tunnel81
 #ip unnumbered Loopback0                                    # LSP source IP address
 #tunnel mode mpls traffic-eng                               # Tunnel mode is MPLS-TE
 #tunnel destination 1.1.1.1                                 # LSP tail-end IP address
 #tunnel mpls traffic-eng autoroute announce                 # install a route in routing table
 #tunnel mpls traffic-eng path-option 1 explicit name 86421  # Path computation CSPF
!
#ip explicit-path name 86421 enable                         # Creating LSP path
 #next-address 6.6.6.6                                      # Strict Next-Hop
 #next-address 4.4.4.4                                      # Strict Next-Hop
 #next-address 2.2.2.2                                      # Strict Next-Hop
 #next-address loose 1.1.1.1                                # Loose Next-Hop
```

Strict next hop means that the IP address must belong to one of my adjacency, whereas, in loose hop the ip address can be to any router not necessary my adjacency router.

3. One LSP using (Explicit | Dynamic) path with End-to-End Protection (One LSP two pre-signaled Paths)

```
#interface Tunnel81
 #ip unnumbered Loopback0
 #tunnel mode mpls traffic-eng
 #tunnel destination 1.1.1.1
 #tunnel mpls traffic-eng autoroute announce
 #tunnel mpls traffic-eng path-option 1 explicit name 86421
 #tunnel mpls traffic-eng path-option protect 1 (explicit name 86521 | Dynamic)
```

The above sample configuration shows a method to perform end-to-end path protection at the head-end. In fact, the head-end pre-establish a protection LSP. This protection LSP is pre-signaled to the tail-end but it is not in use. The head-end will use this protection if the main LSP fails for any reason.

4. One LSP using (Explicit | Dynamic) path with End-to-End Protection (Main path active, backup path standby)

```
#interface Tunnel81
 #ip unnumbered Loopback0
 #tunnel mode mpls traffic-eng
 #tunnel destination 1.1.1.1
 #tunnel mpls traffic-eng autoroute announce
 #tunnel mpls traffic-eng path-option 1 explicit name 86421
 #tunnel mpls traffic-eng path-option 2 explicit name 86521
# tunnel mpls traffic-eng fast-reroute node-protect
```

This LSP tunnel requested a local node protection ,thus, all upstream nodes will try to pre-signal a temporary LSP to the next-next hop.
It is also possible to use only link protection; however, node protection will tolerate faults caused to link and node failure.

5. One LSP using (Explicit | Dynamic) path with Local Protection.

```
#interface Tunnel81
 #ip unnumbered Loopback0
 #tunnel mode mpls traffic-eng
 #tunnel destination 1.1.1.1
 #tunnel mpls traffic-eng autoroute announce
 #tunnel mpls traffic-eng path-option 1 explicit name 86421
 #tunnel mpls traffic-eng path-option 2 explicit name 86521
```

The above sample configuration shows a method to perform end-to-end path protection at the head-end. however, only one LSP is active at a time. The router will choose the path that has lowest preference. If the path with lowest preference fails then router will establish the LSP over the second path.

6. Link Protection at P routers.

Automatic Protection:

```
# mpls traffic-eng auto-tunnel backup
```

This enables the router to automatically establish one or more LSP to protect any LSP that has link protection request.

Manual Protection:

```
# interface GigabitEthernet0/0
 # mpls traffic-eng backup-path tunnel 652
!
# interface Tunnel652
 # ip unnumbered Loopback0
 # tunnel mode mpls traffic-eng
 # tunnel destination 2.2.2.2
 # tunnel mpls traffic-eng path-option 1 explicit name 652
```

As shown above, this method gives control ability over any protected LSP. For example, If any LSP passing interface gig 0/0 is going to fail because node or link failure. The traffic will instantly switchover to tunnel 652.

6.2.2.2 Nesting and Stitching

In many cases, using one end-to-end contiguous LSP is not recommended or not possible. For instance, to make the core network scalable the number of LSPs must be reduced. Thus, stitching and nesting must be performed by core network edge nodes.

In addition, in multi Domains/Areas/ASs, LSP cannot establish if some information is not shared with them, such as tail-end /32 IP address or TED information.

6.2.2.2.1 LSP Stitching

The following points represent the implementation guidelines:

- Fully meshed LSPs between core edge router as shown in the below figure (6.2), R6, R7, R2 and R3, each one of them has two LSPs to the other end core edge router. For example, R6 has two LSP, one terminates at R2 and the second one terminates at R3.

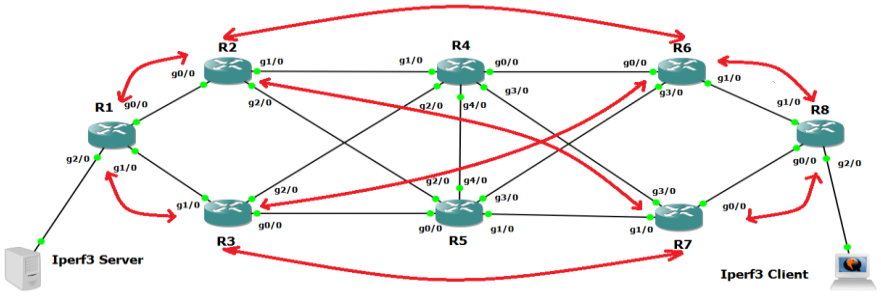


Figure 6.2: LSP Stitching

- LSP configuration is identical to the Contiguous LSP implementation; the LSP path can be specified as strict next hop or loose, it is also possible to rely on CSPF compute each path for each LSP.
- LDP between each head-end and tail-end for each LSP is required. Therefore, LDP and/or TLDP must be enabled in the network. LDP and T-LDP help to perform the stitching. For example, let us assume that PHP is disabled on the network and if Iperf client wants to send data to Iperf server. In this case, R8 does route lookup for Iperf server IP address, then it finds that the next hop is R6 via tunnel 86, the MPLS forwarding table has two labels one for LSP 86 and the second is the LDP binding for Iperf server IP address. Thus, R8 pushes two labels. When the packet reaches R6, R6 will pop the outer label which is the LSP86 label and swap the LDP label based on the local LDP binding and then it pushed a new label that will be used for LSP 62. Once that packet reaches R4, R4 will only swap the outer LSP label and forwarded the packet to R2. R2 will pop the outer LSP label swap LDP label then it pushed AN LSP21 label assuming PHP is disabled. The following is the sample configuration of T-LDP.

```
# mpls ldp neighbor 2.2.2.2 targeted ldp
# mpls ldp neighbor 8.8.8.8 targeted ldp
```

It is important to mention, on order to be able to stitch LSPs, the next hop for the destination must be preferred over the LSP tunnel. To do that, LSP route/link must be advertised with better metric on the local router by using autoroute announce or the entire domain by using forwarding adjacency.

6.2.2.2.2 LSP Nesting:

The following points represent the implementation guidelines:

- The core network routers are meshed like in LSP stitching; R6 and R7 have two LSPs one to R2 and the second to R3. R2 and R3 also have two LSP one to R6 and the second to R7. At each head-end, the LSP link/route must installed on the routing table. These LSP must be configured as MPL-TE links and RSVP must be enabled on them.

```
# interface Tunnel62
 # ip unnumbered Loopback0
 # tunnel mode mpls traffic-eng
 # tunnel mpls traffic-eng forwarding-adjacency
 # tunnel mpls traffic-eng path-option 1 explicit name 642
 # ip rsvp bandwidth                                    # Enabling RSVP
```

- The PEs routers LSPs path are configured with explicit strict next hop, in details, R8's LSP must be configured when strict next hop, so when signaling start the path will be calculated over the core LSP at core LSP head-end as when in the below figure(6.3).

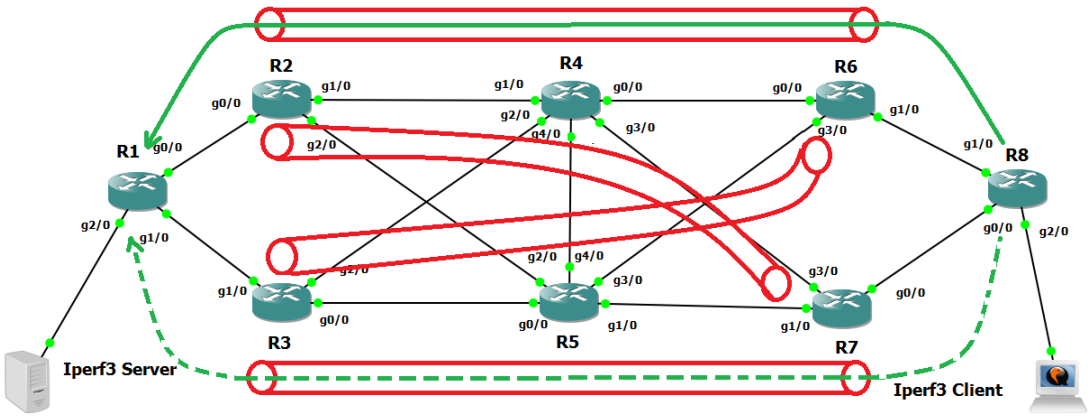
```
#interface Tunnel81
 #ip unnumbered Loopback0
 #tunnel mode mpls traffic-eng
 #tunnel destination 1.1.1.1
 #tunnel mpls traffic-eng autoroute announce
 #tunnel mpls traffic-eng path-option 1 explicit name 8621 verbatim
tunnel mpls traffic-eng path-option protect 1 explicit name 8753 verbatim
!
ip explicit-path name 8621 enable
 next-address ( strict | loose )6.6.6.6
 next-address strict 2.2.2.2
 next-address ( strict | loose )1.1.1.1
```

AS shown in the configuration, next address strict force R6 to used the LSP 82 as. It is also important to use **verbatim** to bypass the topology database verification

Figure 6.3: LSP Nesting

6.2.3 Intra-Area MPL-TE Convergence and Protection Results

Native IP Convergence time



It is worth mentioning that service provider networks must have the capabilities to provide fast layer 3 convergences because everything is built on top layer 3 protocol. For example, to provide fast convergence, fast failure detection must be used. In the lab and Gns3, we tested converge time after link/node failure. The results were not expected, the convergence time was about 6 seconds even with ECMP is being used.

However, in spring 2016, the convergence time was tested on a network that uses Nokia 7750 SR-12 and Juniper SRX-240, the average convergence time for a remote failure was about 600 msec with BFD enabled and about 200 msec with LFA and PFD enabled.

To compute the convergence time between two adjacent routers, we configured all routers to synchronize their local time to central NTP clock, then from the debugging timestamps after a remote failure accrues, we can calculate the time needed for each event as shown below:

- The local router needs about 10 msec to delete the route from the routing table.
- The local router needs about 1 sec to trigger OSPF update.
- The local router needs about 5.532 sec to update its routing table with a new route toward the destination.
- The adjacent router needs 6 sec to install new route.

Local Router

00:05:05.127: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.34.4 on GigabitEthernet1/0 from FULL to DOWN
00:05:05.131: RT: interface GigabitEthernet1/0 removed from routing table
00:05:05.135: RT: delete subnet route to 192.168.24.0/24
00:05:05.139: RT: delete subnet route to 192.168.24.2/32
00:05:06.125 : OSPF-1 FLOOD Gi1/0: Add Type 1 LSA
00:05:10.631: RT: updating ospf 192.168.34.0/24 (0x0): via 192.168.12.1 Gi0/0 1048578
00:05:10.651: RT: updating ospf 20.20.20.0/24 (0x0): via 192.168.12.1 Gi0/0 1048578
00:05:10.659: RT: add 20.20.20.0/24 via 192.168.12.1, ospf metric [110/4]

Adjacent Router

00:05:11.000: RT: closer admin distance for 192.168.24.0, flushing 1 routes
00:05:11.002: RT: del 20.20.20.0 via 192.168.12.2, ospf metric [110/3]

To sum up, it is evident that the router takes about 6 seconds to converge. However, in real implementation for MPLS-TE, IGP convergence time must be less than 1 second.

Contiguous LSP (One LSP , two Active-Active paths):

As mentioned in the implementation, there are one LSP configured at R8, this LSP configured with one main path and a second pre-established path for protection. In another word, it is two LSP from the same head-end to the same tail-end. The main LSP path via R6-R4-R2, and the second LSP path via R7-R5-R3 as shown in the below figure (6.4).

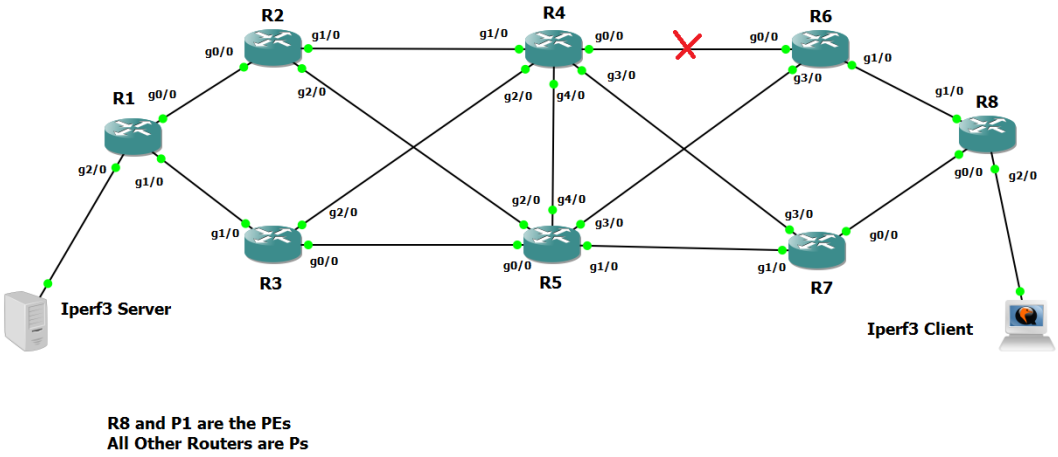


Figure 6.4: contiguous LSP

At the head-end, the LSP is established and the routing table shows the tunnel 81 is the best route toward Iperf server. Once a failure happen anywhere in the path and the head-end receives RSVP path error message, it will switchover to the pre-established second path.

R6 Gi 0/0 Shutdown	R4 Gi 0/0 Shutdown
0.13 sec	**0.20 sec**

Contiguous LSP (One LSP, two Active-Standby paths):

In this case, two paths are configured for LSP 82 the main path is established and the second path will establish after the main path fails.

R6 Gi 0/0 Shutdown	R4 Gi 0/0 Shutdown
0.40 sec	**0.57 sec**

It is clear that the downtime increase because the head-end needs to compute and signal the second path after receiving path error message.

Contiguous LSP (One LSP , two Active-Standby paths with FRR-Node Protection):

In this test, we configured the head-end to ask for FRR node protection from all upstream path nodes. At the upstream node, two types of detours can be used, dynamic detour or a static one. We used a static detour to control the traffic end-to-end.

At R6, LSP tunnel is configured to protect any link/node failure between R6 and R2. In fact, R5 is the detour point. Thus, if failure accrues, R6 will switch the traffic to take R5 to reach R2 and join the main LSP. In addition, R6 will notify the head-end by sending RSVP path error message. The head-end will keep sending the traffic to R6 and will try to establish a second path toward the tail-end.

R6 Gi 0/0 Shutdown	R4 Gi 0/0 Shutdown
No loss	**0.045 sec**

From the above results, R6 will immediately switchover the traffic to the detour because it is local interface failure. However, in the case of interface R4 failure, R6 wait until it detects that the link/node is down before switching the traffic to the detour.

LSP Stitching

As described in the implementation, in this case, the protection is very hard to implement and maintain because the LSP in not end-to-end and many head-ends and tail-ends are involved in the forwarding process. In addition, the head-end and the tail-end must maintain synchronized information not only for the RSVP LSP path but also for the LDP binding.

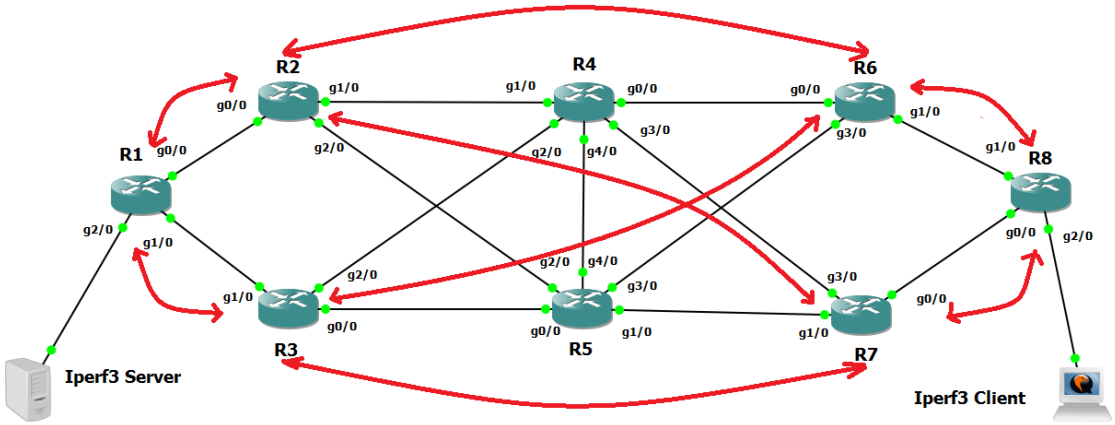


Figure 6.5: LSP Stitching

In this case, each head-end must build an LSP for each possible tail-end at the aggregation layer. In addition, core edge routers must fully mesh. FRR can also be used for temporary detour the traffic way from the failure.

In case FRR is not being used, the recovery time is about 5 seconds this is because the router needs time for deleting the route toward the tail-end and installing the second best route toward the same tail-end.

```
16:53:05.015: RT: interface Tunnel62 removed from routing table
16:53:05.059: RT: del 20.20.20.20 via 0.0.0.0, static metric [1/0]
16:53:05.059: RT: delete subnet route to 20.20.20.20/32
16:53:07.007: RT: interface Tunnel62 removed from routing table
16:53:10.079: RT: del 20.20.20.0 via 2.2.2.2, ospf metric [110/4]
```

Stitched LSP (One LSP, two Active-Active paths with FRR-Node Protection):

R6 Gi 0/0 Shutdown	R4 Gi 0/0 Shutdown
No loss	**0.049 sec**

LSP Nesting:

In this implementation, there are two levels LSPs, level 1 LSP from the PE to the edge core LSR, and level 2 LSP in the core LSRs between all core edges LSRs. Level 1 LSP are signaled inside level 2 LSPs. Therefore, in order to consider the level 2 LSP as TE link, RSVP and traffic engineering must be enabled on them.

With Cisco IOS, advertising level 2 LSPs as TE link by using OSPF or ISIS is not supported. Thus, it is mandatory to use explicit path with strict next hop at the core, and the TED topology database verification must be ignored by using keyword verbatim at the level-1 head-ends.

During implementation and testing, we discover that when level 2 link/node fails, the level-2 head-end sends tear message to the level-1 head-end. The next example will clarify the process.

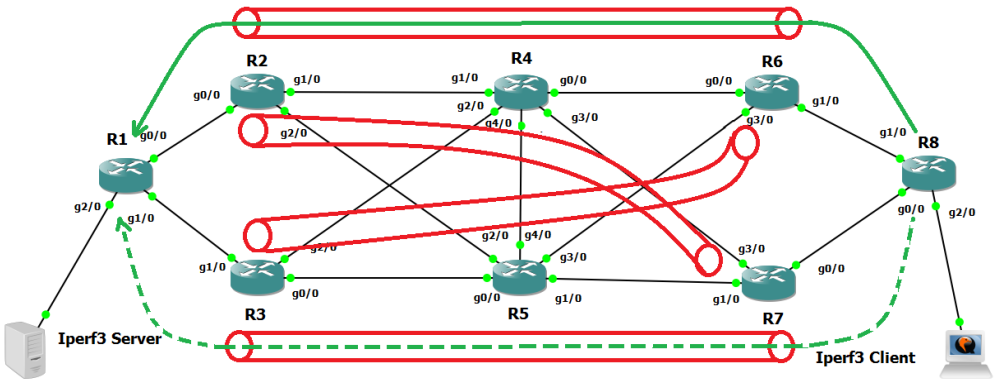


Figure 6.6: LSP Nesting

As shown in the figure(6.6), level 1 LSP is configured from the head-end R8 to the tail-end R1, at the head-end, the path to R6 can be set strict or loose and the path from R2 to the tail-end R1 can also be set as strict or loose. However, in the core, the path next hop must set to the level 2 tail-end which is R2 in this example.

The path from level 1 head-end R8 to the level 2 head-end R6, and the path from level-2 tail-end R2 to level 1 tail-end R1 can be protected with FRR and we got the same previous result. However, FRR at level 2 LSP is behavior differently, we noticed that we there is a failure in the core network, the traffic is switched over to the detour LSP, but R6 the level 2 head-end send RSVP reservation tear message to leve-1 head-end R8. Once R8 receives the message, it sends path tear message to R1 the level 1 tail-end. This process causes re-signalling the LSP over the detour.

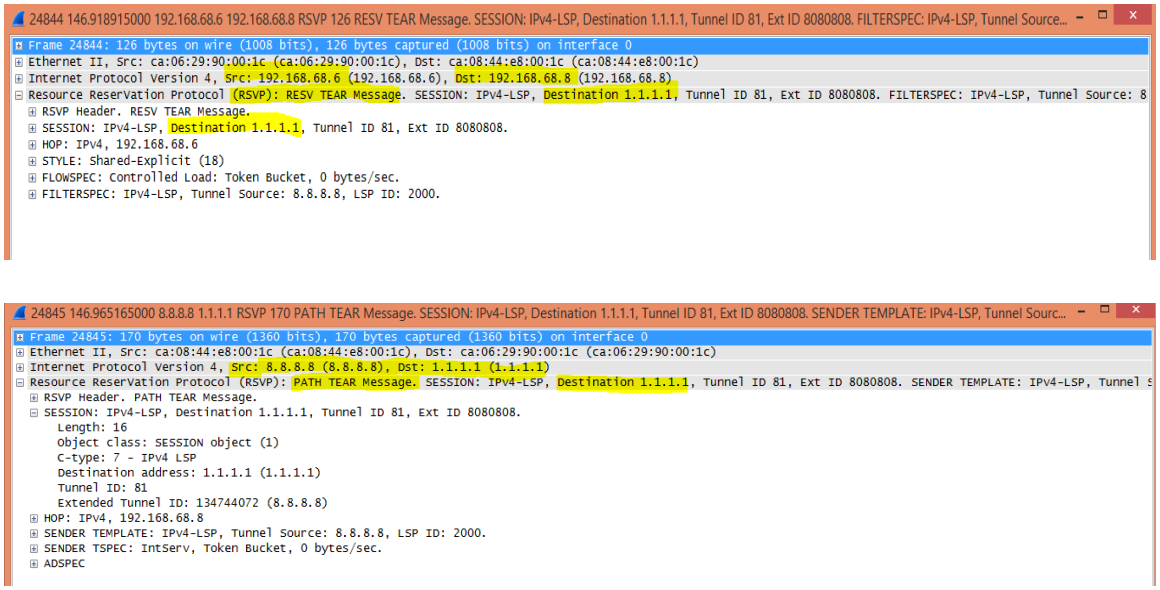


Figure 6.7: RSVP Tear Message

Nested LSP (One LSP, two Active-Active paths with FRR-Node Protection):

In this scenario, as shown in the previous diagram, R8 the head-end has two active-active Paths towards the tail-end. The main path is nested inside LSP 62 and the secondary path is nested inside LSP 73. In the case of level 2 LSP 62 failure, the R6 will protect the LSP 62 with local protection detour terminates at the tail-end. In addition, R6 will send tear message to the head-end. This message will cause the head-end to switch to the secondary path. The downtime for this test was similar to the previous cases, as shown in the next table.

R6 Gi 0/0 Shutdown	R4 Gi 0/0 Shutdown
No loss	**0.048 sec**

6.3 Inter-Area Traffic Engineering

In this section, we will implement and evaluate Inter-Area MPLS-TE LSP. As described in chapter 4, the issue of establishing an MPLS-TE LSP is that areas have OSPF topology, thus, path computation has some restriction. The following subsection illustrates three methods of path computation.

6.3.1 Inter-Area MPL-TE Implementation:

- The same topology is being used as in Intra-Area Traffic engineering. However, R2,R3,R6 and R7 are the Area border routers, as shown below:

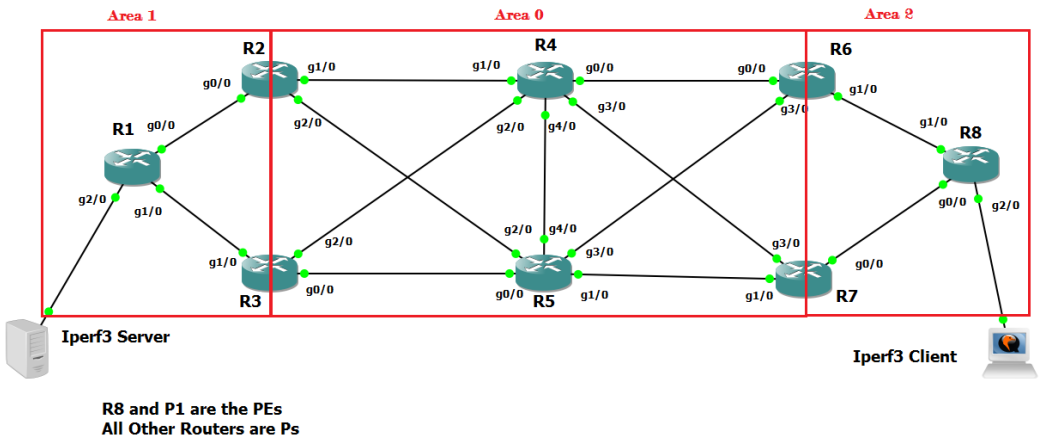


Figure 6.8: Inter-Area Lab Topology

- IP Plan: the subnet 192.168.0.0/16 is being used with the same design as in Intra-Area Traffic engineering. In addition, the router number in all octets represents each router loopback address.
- L2: All interfaces are 1G-Ethernet.
- L3: OSPFv2 is the IGP, tree OSPF area is being used. The same Intra-Area configuration applied, but the ABRs interfaces must be configured to reflect the area they belong to as shown below:

```
R2#                                 # ABR-2
#interface GigabitEthernet0/0
 #ip ospf 1 area 1                  # enable OSPF for Area 1
 #ip ospf network point-to-point    # for fast ospf neighborship formation.
!
 #interface GigabitEthernet1/0
 #ip ospf 1 area 0                  # enable OSPF for Area 0
 #ip ospf network point-to-point    # for fast ospf neighborship formation.
!
#interface GigabitEthernet2/0
 #ip ospf 1 area 0                  # enable OSPF for Area 0
 #ip ospf network point-to-point    # for fast ospf neighborship formation.
```

- MPLS-TE: MPLS-TE configuration is the same as Intra-Area, however, for the ABRs, the Traffic engineering for all Area attached to each ABR must be addressed to OSPF global configuration, as shown below:

```
#mpls traffic-eng tunnels
 #interface GigabitEthernet0/0
 #mpls traffic-eng tunnels #Enable TE-RSVP on the interface
 #ip rsvp bandwidth          #Assignee BW to be used by LSP. The default is 75% of the physical BW
!
#router ospf 1
 #mpls traffic-eng router-id Loopback0 # Source IP for MPLS-TE
 #mpls traffic-eng area 0              # OSPF area that shares TED information inside area 0
 #mpls traffic-eng area 1              # OSPF area that shares TED information inside area 1
```

- Configure MPLS-TE LSP tunnel: In this LAB an LSP tunnel is configured from R8 to R1, multiple LSP setups and path calculations are being used as described in the following points:

6.3.1.1 Contiguous LSP (End-to-End ERO Expansion):

In multi OSPF area, it is not possible to establish End-to-End LSP by using dynamic path computation, because areas do not share topology information. Thus, the head-end cannot calculate the path to the head-end. However, there are two methods to establish end-to-end LSP as listed below:

- ERO Expansion: in this method, the path is calculated per area. In details, the head-end compute the path to one of its ABRs, then the ABR expand the path to the next ABR or the tail-end, then the new sub-path is added to the ERO. The rule is to define the ABR address as loose hops at the head-end. The following is the head-end R8 sample configuration:

```
#interface Tunnel81
 #ip unnumbered Loopback0                                        # LSP source IP address
 #tunnel mode mpls traffic-eng                                   # Tunnel mode is MPLS-TE
 #tunnel destination 1.1.1.1                                     # LSP tail-end IP address
 #tunnel mpls traffic-eng autoroute announce                     # install a route in routing table
 #tunnel mpls traffic-eng path-option 1 explicit name 86421 # Path computation ERO Expansion
!
#ip explicit-path name 86421 enable                             # Creating LSP path
 #next-address loose 6.6.6.6                                    # use CSPF for path computation
 #next-address loose 2.2.2.2                                    # use CSPF for path computation
 #next-address loose 1.1.1.1                                    # use CSPF for path computation
```

The head-end R8 uses CSPF to compute the path to ABR6, then ABR6 used CSPF to expand the path to ABR2, then ABR2 expands the path to the head-end.
Each ABR adds the new computed path to the ERO, and send it downstream until it reaches the head-end.

- Explicit Strict Hops: It is also possible to specify all hops end-to-end, but in order for this to work, topology database verification must be disabled by using keyword verbatim.

```
#interface Tunnel81
 #ip unnumbered Loopback0
 #tunnel mode mpls traffic-eng
 #tunnel destination 1.1.1.1
 #tunnel mpls traffic-eng autoroute announce
 #tunnel mpls traffic-eng path-option 1 explicit name 86421 verbatim # Path computation explicit
!
#ip explicit-path name 86421 enable
 #next-address 6.6.6.6
 #next-address 4.4.4.4
 #next-address 2.2.2.2
 #next-address 1.1.1.1
```

Verbatim is used to bypass the topology database verification.

6.3.1.2 Nesting and Stitching (Per domain Path Computation):

Nesting and stitching LSPs are another ways to compute and signal the path from the head-end to the tail-end. The implementations of these methods are exactly the same as we described in single OSPF area.

6.3.1.3 Path Computation Element (PCE)

In this scenario, The same topology has being used. However, we replaced the head-end R8 and the ABRs (R7,R6,R3,R2) with Cisco IOS-XR V 5.3.2, because it is the only platform that supports PCE implementation.

Implementing PCE on IOS-XR platforms

- At the head-end (PCC), the path-option must be configured to use PCE dynamic path calculation.

```
#interface tunnel-te81
 #ipv4 unnumbered Loopback0
 #autoroute destination 20.20.20.1
 #destination 11.11.11.11
 #fast-reroute protect node
 #path-option 1 dynamic pce                                  # Backup path. This tunnel uses PCE for path
                                                               computation.
 #path-option 2 dynamic pce address ipv4 77.77.77.77 # Backup path. This tunnel uses PCE for path
                                                       computation.
```

This LSP uses dynamic PCE path computation. The keyword dynamic pce enable auto discovery for any PCE peer in the area.

- At each ABR, PCE must be enabled as shown in the bellow:

```
#mpls traffic-eng
 #Pce
  #address ipv4 22.22.22.22 # this is local address, this router will announce itself as PCE peer. PCE
                                discovery is done by using OSPF routing protocol.
```

It is also possible to use static PCE to PCE peering by using the following command.

```
#mpls traffic-eng
 #pce
  #peer ipv4 66.66.66.66    # this is the remote PCE peer.
```

PCC and PCE request and reply messages

In this implementation, the head-end (Path Computation Client) sends a path computation to request the closet (Path Computation Element), many PCEs can be used as in our topology, the LSP must be calculated over 3 areas. The following points explain the process:

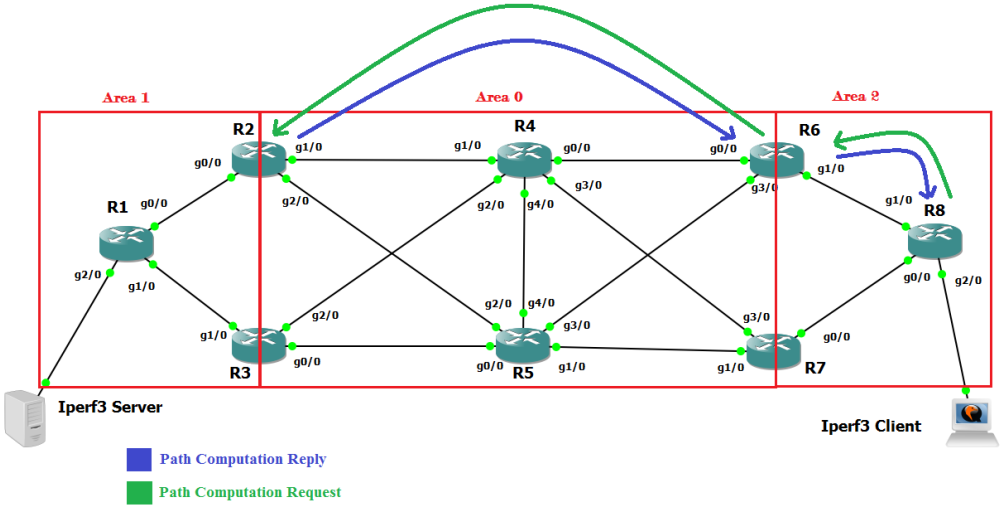


Figure 6.9: Path Computation Element

- R8 send path computation request to its PCE peer R6.

```
Frame 34: 130 bytes on wire (1040 bits), 130 bytes captured (1040 bits) on interface 0
Ethernet II, Src: CadmusCo_fb:71:f0 (08:00:27:fb:71:f0), Dst: CadmusCo_56:df:19 (08:00:27:56:df:19)
Internet Protocol Version 4, Src: 88.88.88.88 (88.88.88.88), Dst: 66.66.66.66 (66.66.66.66)
Transmission Control Protocol, Src Port: 33982 (33982), Dst Port: 4189 (4189), Seq: 13, Ack: 13, Len: 76
Path Computation Element communication Protocol
  PATH COMPUTATION REQUEST MESSAGE Header
  RP object
  END-POINT object
     Object Class: END-POINT OBJECT (4)
     0001 .... = Object Type: 1
   Flags
     Object Length: 12
     Source IPv4 Address: 88.88.88.88 (88.88.88.88)
     Destination IPv4 Address: 11.11.11.11 (11.11.11.11)
  LSPA object
  BANDWIDTH object
  METRIC object
```

Figure 6.10: PCC to PCE Request Message

- R6 checks the request and if the tail-end IP address originated is not in its own area, R6 sends path computation request to its PCE peer R2.

```
Frame 50: 130 bytes on wire (1040 bits), 130 bytes captured (1040 bits) on interface 0
Ethernet II, Src: CadmusCo_dd:a9:6d (08:00:27:dd:a9:6d), Dst: ca:04:35:38:00:08 (ca:04:35:38:00:08)
Internet Protocol Version 4, Src: 66.66.66.66 (66.66.66.66), Dst: 22.22.22.22 (22.22.22.22)
Transmission Control Protocol, Src Port: 53843 (53843), Dst Port: 4189 (4189), Seq: 9, Ack: 9, Len: 76
Path Computation Element communication Protocol
  PATH COMPUTATION REQUEST MESSAGE Header
  RP object
  END-POINT object
     Object Class: END-POINT OBJECT (4)
     0001 .... = Object Type: 1
   Flags
     Object Length: 12
     Source IPv4 Address: 88.88.88.88 (88.88.88.88)
     Destination IPv4 Address: 11.11.11.11 (11.11.11.11)
  LSPA object
  BANDWIDTH object
  METRIC object
```

Figure 6.11: PCE to PCE Request Message

- R2 checks the request and it finds that the tail-end information in the local TED. Thus, R2 send path computation reply to R6. This message contains every possible path that this LSP can be used.

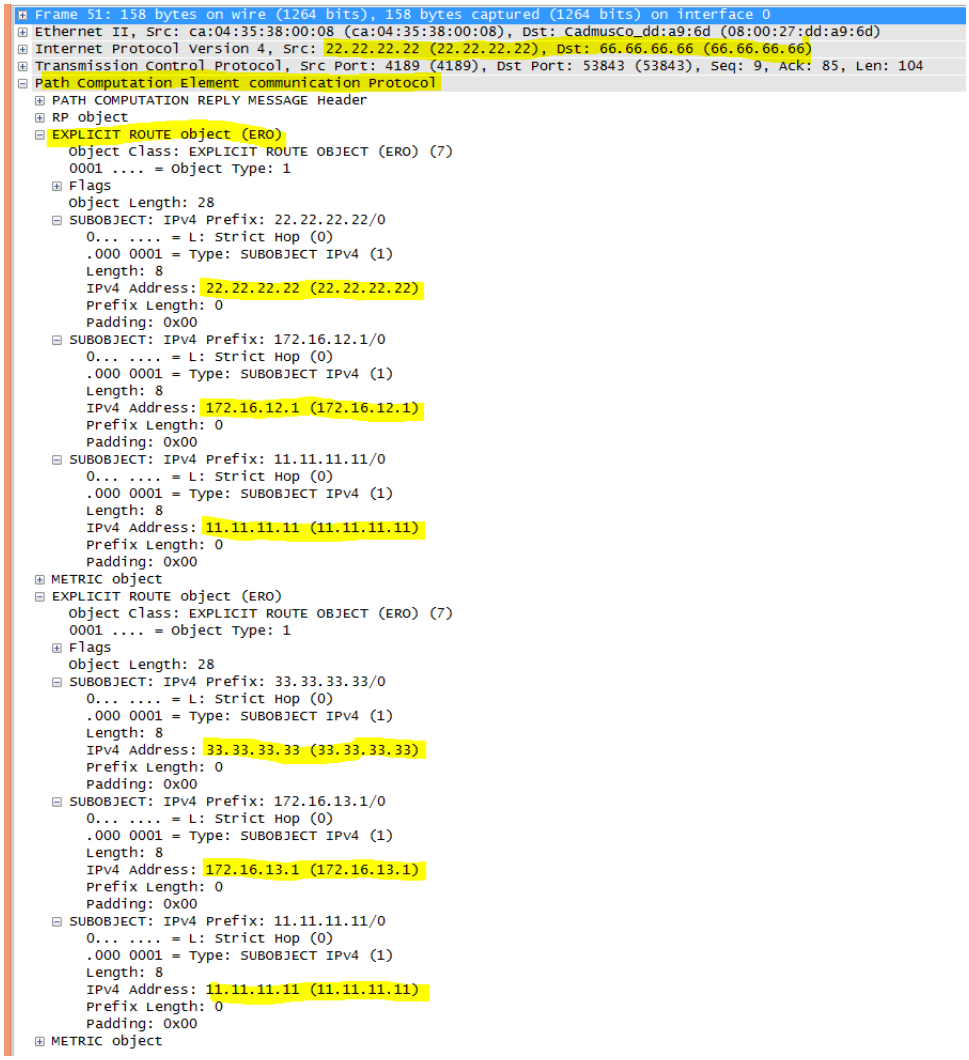


Figure 6.12: PCE to PCE Reply Message

- R6 check the message and chose the entire path from the head-end to the tail-end, then R6 sends the computed path to the head-end.

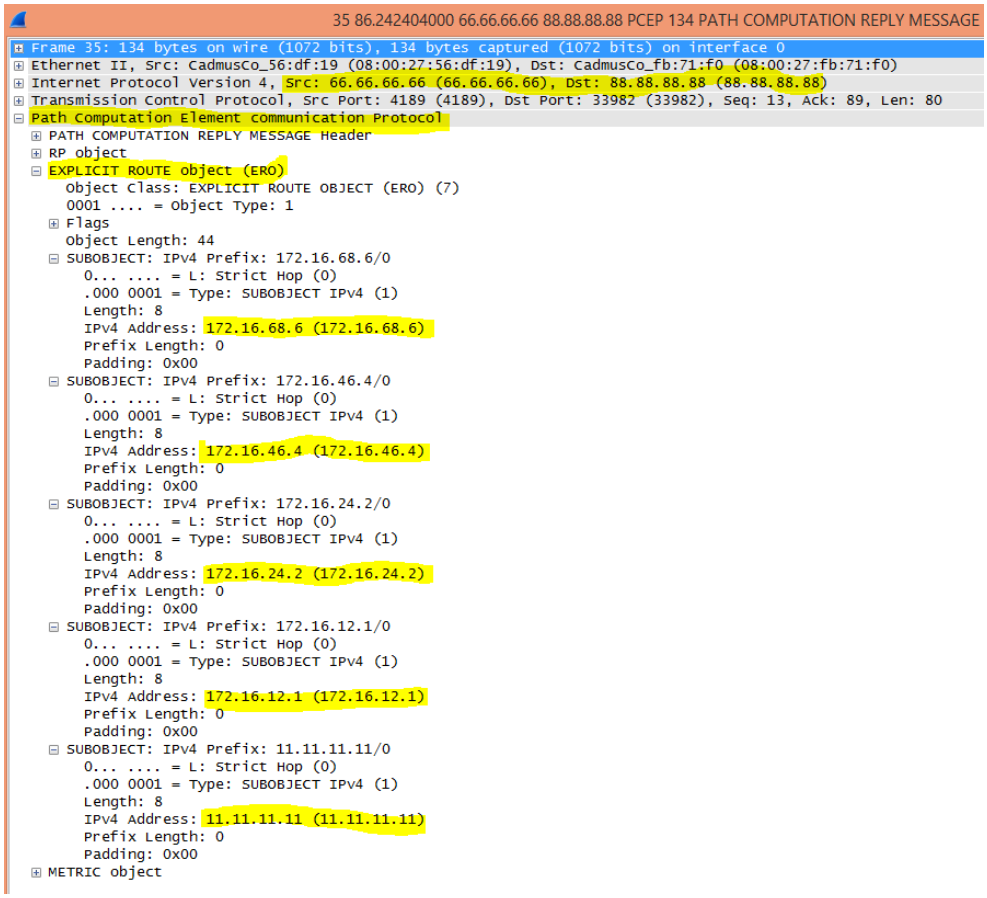


Figure 6.13: PCE to PCC Reply Message

6.3.2 Inter-Area MPL-TE Convergence and Protection Results

Path computation is the only difference between single OSPF domain and multiple OSPF domains because areas do not share topology information between each other.

Contiguous LSP (End-to-End):

ERO Expansion, Contiguous LSP (two Active-Standby paths with FRR-Node Protection):

In this scenario, we have configured the-head end to use ERO expansion for two paths, the primary path will be used, and in the case of failure, R6 will use the detour via R5.

R6 Gi 0/0 Shutdown	R4 Gi 0/0 Shutdown
No loss	**0.044 sec**

LSP Stitching (End-to-End):

This scenario is the same as in Intra-Area Traffic Engineering.

Stitched LSP (One LSP, two Active-Active paths with FRR-Node Protection):

R6 Gi 0/0 Shutdown	R4 Gi 0/0 Shutdown
No loss	**0.055 sec**

LSP Nesting (End-to-End):

This scenario is the same as in Intra-Area Traffic Engineering.

Nested LSP (One LSP, two Active-Active paths with FRR-Node Protection):

R6 Gi 0/0 Shutdown	R4 Gi 0/0 Shutdown
No loss	**0.047 sec**

Path Computation Element (PCE):

As explained before, in this scenario, the head-end consults PCE to compute the path end-to-end.

One LSP (One LSP, two Active-Standby paths using PCE with FRR-Node Protection):

R6 Gi 0/0 Shutdown	R4 Gi 0/0 Shutdown
No loss	**0.040 sec**

6.4 Inter-AS L3VPNs

As described in chapter five, In Intra-As L3VPN need MP-IBGP peering between the PEs where the customer sites reside. However, if a customer needs L3VPN between two sites reside in different AS, MP-IBGP between these PEs are not possible. In fact, there are three alternative solutions to implemented Inter-AS L3VPN, the following sections cover the implementation for each Inter-AS method.

6.4.1 Inter-AS L3-VPN Option A (Back-to-Back VRF)

In this method, each AS treat the adjacent AS as L3-VPN customer by using back-to-back VRF as shown in the next figure (6.14).

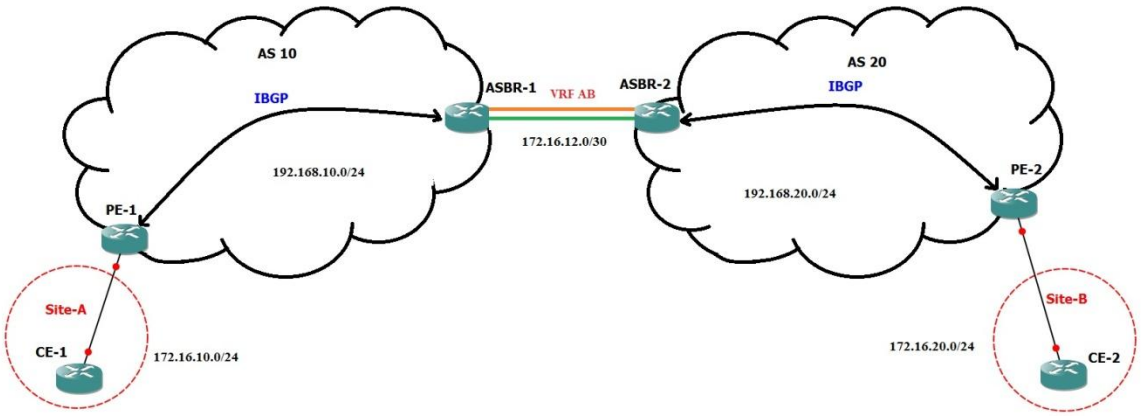


Figure 6.14: Inter-AS L3-VPN Option A Topology

6.4.1.1 Inter-AS L3-VPN Option A Implementation

- OSPFv2 is the routing protocol. Each AS has separate OSPF domain. OSPFv2 configuration is exactly as shown previously in this chapter.
- MPLS-LDP or MPLS-TE must be enabled in each AS separately. MPLS-TE configuration is exactly as shown previously in this chapter. In addition, to configure LDP the following commands are required under each core interface.

```
# interface GigabitEthernet1/0
 # mpls label protocol ldp        # enabling LDP protocol
 # mpls ip                        # enabling mpls
```

- AS10, PE1 L3-VPN, each customer interface must be mapped to its VRF.

```
# ip vrf Site-A
 # rd 10:10                       # Add 10:10 to the customer routes
 # route-target export 10:10 # Export all route to any vrf that has route-target import 10:10
 # route-target import 10:10 # Import any route that has route target export 10:10 to this vrf
!
# interface GigabitEthernet0/0
 # ip vrf forwarding Site-A
 # ip address 172.16.10.1 255.255.255.252
!
ip route vrf Site-A 172.16.10.0 255.255.255.0 172.16.10.2 # Static route pointing to customer network
```

- AS20, PE2 L3-VPN, each customer interface must be mapped to its VRF.

```
# ip vrf Site-B
 # rd 20:20                       # Add 10:10 to the customer routes
 # route-target export 20:20 # Export all route to any vrf that has route-target import 20:20
 # route-target import 20:20 # Import any route that has route target export 20:20 to this vrf
!
# interface GigabitEthernet0/0
 # ip vrf forwarding Site-B
 # ip address 172.16.20.1 255.255.255.252
!
# ip route vrf Site-B 172.16.20.0 255.255.255.0 172.16.20.2 # Static route pointing to customer network
```

- AS10, ASBR1 L3VPN

```
# ip vrf Site-A
 # rd 10:10                          # Add 10:10 to the customer routes
 # route-target export 10:10 # Export all route to any vrf that has route-target import 20:20
 # route-target import 10:10 # Import any route that has route target export 20:20 to this vrf
!
# interface GigabitEthernet0/0.10 # Back-to-Back VRF
 # encapsulation dot1q 10
 # ip vrf forwarding Site-A
 # ip address 192.168.12.1 255.255.255.252
 # mpls label protocol ldp
 # mpls ip
!
# ip route vrf Site-A 172.16.20.0 255.255.255.0 192.168.12.2 # Static route pointing to Site-B customer network.
```

- AS20, ASBR2 L3VPN

```
# ip vrf Site-B
 # rd 20:20                          # Add 10:10 to the customer routes
 # route-target export 20:20 # Export all route to any vrf that has route-target import 20:20
 # route-target import 20:20 # Import any route that has route target export 20:20 to this vrf
!
# interface GigabitEthernet0/0.10  # Back-to-Back VRF
 # encapsulation dot1q 10
 # ip vrf forwarding Site-B
 # ip address 192.168.12.2 255.255.255.252
 # mpls label protocol ldp
!
ip route vrf Site-B 172.16.10.0 255.255.255.0 192.168.12.1 # Static route pointing to Site-A customer
                                                            network.
```

- AS 10 MP-IBGP

PE-1

```
#router bgp 10
 #bgp log-neighbor-changes
 #neighbor 192.168.10.255 remote-as 10                  # enabling IBGP peering with ASBR loopback 0
 #neighbor 192.168.10.255 update-source Loopback0  # use loopback 0 as source address for IBGP seesion
 !
 #address-family vpnv4                                  # enabling VPNV4 label exchange with ASBR1
  #neighbor 192.168.10.255 activate
  #neighbor 192.168.10.255 send-community extended
 #exit-address-family
 !
 #address-family ipv4 vrf Site-A                        # enabling Site-A route distribution to ASBR1
  #redistribute static
 #exit-address-family
```

ASBR-1

```
#router bgp 10
 #bgp log-neighbor-changes
 #neighbor 192.168.10.254 remote-as 10                  # enabling peering with ASBR loopback 0
 #neighbor 192.168.10.254 update-source Loopback0 # use loopback 0 as source address for IBGP session
 !
 #address-family vpnv4                                  # enabling VPNV4 label exchange with ASBR1
  #neighbor 192.168.10.254 activate
  #neighbor 192.168.10.254 send-community extended
 #exit-address-family
 !
 #address-family ipv4 vrf Site-A                        # enabling Site-A route distribution to
  #redistribute static
 #exit-address-family
```

- AS 20 MP-IBGP

PE-2

```
#router bgp 20
 #bgp log-neighbor-changes
 #neighbor 192.168.20.255 remote-as 20                  # enabling IBGP peering with ASBR loopback 0
 #neighbor 192.168.20.255 update-source Loopback0  # use loopback 0 as source address for IBGP session
 !
 #address-family vpnv4                                  # enabling VPNV4 label exchange with ASBR1
  #neighbor 192.168.20.255 activate
  #neighbor 192.168.20.255 send-community extended
 #exit-address-family
 !
 #address-family ipv4 vrf Site-B                        # enabling Site-B route distribution to ASBR1
  #redistribute static
 #exit-address-family
```

ASBR-2

```
#router bgp 20
 #bgp log-neighbor-changes
 #neighbor 192.168.20.254 remote-as 20                  # enabling peering with ASBR loopback 0
 #neighbor 192.168.20.254 update-source Loopback0 # use loopback 0 as source address for IBGP seesion
 !
 #address-family vpnv4                                  # enabling VPNV4 label exchange with ASBR1
  #neighbor 192.168.20.254 activate
  #neighbor 192.168.20.254 send-community extended
 #exit-address-family
 !
 #address-family ipv4 vrf Site-B                        # enabling Site-B route distribution to
  #redistribute static
 #exit-address-family
```

6.4.1.2 Traffic Forwarding Between Sites

Let us assume that site-B want to communicate with site-A. In this case, CE-2 sends packets to CE-1, once the packet arrives at PE-2. PE-2 checks the forwarding table for VRF-Site B, then it pushes two labels the inner label is a VPNV4 label learned from ASBR-2 via BGP, and the outer label is the transport label (LDP or RSVP label). This outer label is swapped at each hop inside AS20. The last LSR before the ASBR-2 pops the outer label if PHP is been used. Once the packet arrives at ASBR-2, ASBR-2 pops the last label and pushes LDP label and forward the packet to ASBR-1 (if LDP in not used on the sub-interface between ASs, then the packet is forwarded with

no label in the label stack). Once the packet is received by ASBR-1, ASBR-1 check the forwarding table and pushes two labels, the inner label is the VPNV4 label and the outer label is the transport label, the transport label is treated as described in AS 20. Once the packet arrives at PE-1, PE-1 pops the last label and forwards the packet to CE-1.

6.4.2 Inter-AS L3-VPN Option B Implementation

Inter-AS option B uses VPNV4 EBGP peering between ASBRs. In this implementation, we have used the same topology as shown in option A implementation. To avoid repetition the most of the option A configuration will be used with option B. The following points illustrate the differences and the new configuration needed.

- The VRFs belong to the same L3VPN client must be configured the same route-target. This is because the VPNV4 routes are maintained end-to-end.
- AT the ASBRs, the only configuration needed is VPNV4 EBGP.

ASBR-1

```
# router bgp 10
 #  bgp log-neighbor-changes
 # no bgp default route-target filter            # keep all VPNV4 routes even if there is no local vrf for them
 # neighbor 192.168.10.254 remote-as 10
 # neighbor 192.168.10.254 update-source Loopback0
 # neighbor 192.168.12.2 remote-as 20
  !
 # address-family vpnv4
 # neighbor 192.168.10.254 activate
 # neighbor 192.168.10.254 send-community extended
 # neighbor 192.168.10.254 next-hop-self  # change the next hop to for VPNV4 to the local IP address
 # neighbor 192.168.12.2 activate          # VPNV4 peering with adjacent AS
 # neighbor 192.168.12.2 send-community extended
 # exit-address-family
```

ASBR-2

```
# router bgp 20
 #  bgp log-neighbor-changes
 # no bgp default route-target filter          # keep all VPNV4 routes even if there is no local vrf for them
 # neighbor 192.168.20.254 remote-as 20
 # neighbor 192.168.20.254 update-source Loopback0
 # neighbor 192.168.12.1 remote-as 10
  !
 # address-family vpnv4
 # neighbor 192.168.20.254 activate
 # neighbor 192.168.20.254 send-community extended
 # neighbor 192.168.20.254 next-hop-self  # change the next hop to for VPNV4 to the local IP address
 # neighbor 192.168.20.2 activate          # VPNV4 peering with adjacent AS
 # neighbor 192.168.12.1 send-community extended
 # exit-address-family
```

- It is very important to configure each ASBR to accept all VPNV4 route even if there is no local VRF for that route. This is mandatory for a successful VPNV4 routes exchange. In addition, each ASBR must change the next hop address to itself before advertising VPNV4 route to its local IBGP peer. This because the default next hop is the adjacent ASBR and the normal behavior is not to advertise the adjacent IP address to the local IGP.
- The last important point is to configure the interfaces between ASBR1 and ASB2 to use BGP transport label by using the following command.

ASBR-1

```
# interface GigabitEthernet0/0
 # ip address 192.168.12.1 255.255.255.0
 # mpls bgp forwarding                    #  Enabling bgp MPLS forwarding
```

ASBR-2

```
# interface GigabitEthernet0/0
 # ip address 192.168.12.2 255.255.255.0
 # mpls bgp forwarding                    #  Enabling bgp MPLS forwarding
```

6.4.2.1 Traffic forwarding between sites

Option B traffic forwarding from CE-2 to CE-1 is similar to option A. In fact, both use the same transport mechanism. The only different is that CE-2 has VPNV4 label that advertised by PE-1. Thus, the VPNV4 inner label is maintained end-to-end.

6.4.3 Inter-AS L3-VPN Option C Implementation

As explained in chapter five, Inter-AS option C is the best option because it removes the overhead from ASBR. In this method, muli-hop EBGP is needed between the PEs. In addition, route leaking between ASs is compulsory because the next hop for the VPNV4 is the another end PE. Furthermore, in addition, End-to-End LSP is also needed.

LSP passing multiple AS can be implemented by using different methods, for example, Path Computation Element can be used for path computation. As for signaling methods, LSP nesting and stitching can be used. In this implementation, we have used LSP stitching be using BGP

label unicast (BGP-LU). The following points illustrate the option C implementation on the same topology as in option A and B:

- PE-1 and PE-2 have VPNV4 Ebgp peering by using multi-hop Ebgp as shown below:

PE-1

```
# router bgp 10
 # bgp log-neighbor-changes
 # neighbor 192.168.20.254 remote-as 20          # Enabling e-bgp MPLS forwarding
 # neighbor 192.168.20.254 ebgp-multihop 250     # allow up to 250 hop
 # neighbor 192.168.20.254 update-source Loopback0
 !
 # address-family vpnv4
  # neighbor 192.168.20.254 activate
  # neighbor 192.168.20.254 send-community extended
  # exit-address-family
 !
 # address-family ipv4 vrf TEST
  # redistribute connected
  # address-family vpnv4
```

PE-2

```
# router bgp 20
 # bgp log-neighbor-changes
 # neighbor 192.168.10.254 remote-as 10          # Enabling e-bgp MPLS forwarding
 # neighbor 192.168.10.254 ebgp-multihop 250     # allow up to 250 hop
 # neighbor 192.168.10.254 update-source Loopback0
 !
 #address-family vpnv4
 # neighbor 192.168.10.254 activate
 # neighbor 192.168.10.254 send-community extended
 # exit-address-family
 !
 # address-family ipv4 vrf TEST
 # redistribute connected
 # address-family vpnv4


# interface GigabitEthernet0/0
 # ip address 172.16.12.2 255.255.255.0
 # mpls bgp forwarding                    # Enabling bgp MPLS forwarding
```

- ASBR -1 and ASBR-2 have IPV4 Ebgp peering. In addition, the address that being used for PE to PE Ebgp peering must advertise via BGP, later each router redistribute these

address into the IGP. It also important to enable BGP-LU by using BGP keyword send-label, as shown in the configuration below:

ASBR-1

```
# router bgp 10
 # bgp log-neighbor-changes
 # no bgp default route-target filter
 # network 192.168.10.254 mask 255.255.255.255  # Advertising PE-1 IP address to AS 20
 # neighbor 192.168.12.2 remote-as 20
 # neighbor 192.168.12.2 send-label         # allow label exchange and mapping  for the network advertized
!
# router ospf 10
 # redistribute bgp 10 metric 10 subnets      # Inject received route into OSPF domain
```

ASBR-2

```
# router bgp 20
 # bgp log-neighbor-changes
 # no bgp default route-target filter
 # network 192.168.20.254 mask 255.255.255.255 # Advertising PE-2 IP address to AS 20
 # neighbor 192.168.13.1 remote-as 10
 # neighbor 192.168.12.1 send-label          # allow label exchange and mapping  for the network advertized
!
# router ospf 20
 # redistribute bgp 20 metric 10 subnets      # Inject received route into OSPF domain
```

6.4.3.1 Traffic forwarding between sites

In this method, CE-1 sends the traffic to PE-1, PE-1 pushes two labels, the inner label is the VPNV4 label that has been received from PE-2 over EBGP peering and the outer label for the transport. Since this implementation uses end-to-end LSP, the transport label is swapped by each LSR in the path. Once the packet arrives at the node before PE-2, the outer transport label is popped out and packet forwarded only with the VPNV4 label, once the packet arrives at PE-2 , PE-2 pops the label and forward the packet to CE-2

6.5 Discussions and Evaluation

6.5.1 Intra-Area Traffic Engineering:

Based on the theoretical research and LAB results, it is very recommended to use single domain/Area for Traffic Engineering for the following reasons.

- The head-end has completed traffic engineering database view. Therefore, the head-end is capable of computing an LSP path end-to-end from the head-end to the tail-end.
- The path from the head-end to the tail-end is always optimal.
- The LSP constraints are maintained in the entire domain, there is no extra overhead needed for constraints translation.
- Local protection protects the main LSP with downtime less than 50 msec.

6.5.2 Intre-Area Traffic Engineering:

Inter-Area Traffic Engineering becomes a mandatory requirement as the network grows and network scalability is not maintained. As we have seen in the theoretical research and LAB results, path computation for an LSP that satisfies the LSP constraints is difficult, as the LSPs must pass multiple areas; the following points are the challenges Intra-Area Traffic Engineering encounter.

- The head-end has partial TED view, this limitation forces an LSP computation to be computed per area basis. Thus, first, the LSP establishment is not a guarantee; Second LSP constrained may not maintained end-to-end.
- Relatively longer time is needed to compute and signal an LSP.
- The head-end may not be informed about a failure on the LSP path.
- The computed path may be optimal in each area, but the overall path may be suboptimal.
- At the head-end pre-established backup path is needed especially with the LSP is nested inside an LSP.

It is clear that the main issue with Inter-Area MPLS-TE is path computation. Thus, it is highly recommended to use path computation element. PCE has a full overview about all LSP and TE links in all domains.

6.5.3 Inter-AS L3-VPN Traffic Engineering:

As we discussed there are three options to implement Inter-AS L3VPN, option A, option B, and option C. Obviously, option C is the best option. With option A (back-to-back VRF) the ASBRs, must maintain a VRF table per L3VPN customer. Option B is a better option that A because Option B is only maintained VPNV4 at the ASBRS. Option C, however, the ASBRs are not involved in the VPN process, because multi-hop Ebgp is established between the PEs in different ASs. The only requirement for option C to work is that the ASs must agree to lead the IP address of the PEs.

6.5.4 LSP Signaling

As we have illustrated there are three methods to signal and LSP, contiguous LSP, LSP stitching and LSP nesting. Contiguous LSP is the best option is the number of LSPs passing the core is limited. The number is large and there is scalability issue to maintain these LSPs then, LSP stitching or/and LSP nesting must be used, however, LSP nesting has an issue with local protection, as it mentioned in the next point.

- In the LAB with LSP nested implementation, we noticed that Level 1 LSP cannot be protected inside the core LSP (level 2 LSP). The only option to protect level 1 LSP is to protect level 2 LSP. However, during protection test, we noticed that the protection works but the level 2 head-end does not suppress tear-down message for level 1 LSP. Thus, during failure and while the traffic is going over the detour, the level-1 is re-established over the detour LSP which lead to traffic drops during the signaling process. To overcome this issue we configure the head-end to pre-establish backup path and once the head-end receives tear-down message the traffic in immediately switched over the backup path

REFERENCES

- **Minei, I., & Lucek, J. (2008). MPLS-enabled applications: Emerging developments and new technologies. Chichester, England: J. Wiley & Sons.**
- **Tadimety, P. R. (2015). OSPF: A Network Routing Protocol. Berkeley, CA: Apress.**
- **Moy, J. T. (1998). OSPF: Anatomy of an Internet routing protocol. Reading, MA: Addison-Wesley.**
- **Pepelnjak, I., & Guichard, J. (2002). MPLS and VPN architectures. Indianapolis, IN: Cisco Press.**
- **Ghein, L. D. (2007). MPLS fundamentals. Indianapolis, IN: Cisco Press.**
- **Guichard, J., Pepelnjak, I., & Apcar, J. (2003). MPLS and VPN Architectures, Volume II. Cisco Press.**
- **L. Andersson, I. Minei and B. Thomas, *Experience with the Label Distribution Protocol (LDP)*, RFC 5037, October 2007**
- **M. Jork, A. Atlas and L. Fang, LDP IGP Synchronization, RFC 5443, March 2009**
- **R. Aggarwal, D. Papadimitriou, S. Yasukawa (eds), Extensions to Resource Reservation Protocol-Traffic Engineering (RSVP-TE) for Point-to-Multipoint TE Label Switched Paths (LSPs), RFC 4875, May 2007**
- **D. Awduche, J. Malcolm, J. Agogbua, M. O'Dell, and J. McManus, Requirements for Traffic Engineering over MPLS, RFC2702, September 1999.**
- **Farrel, J. Vasseur and J. Ash, Path Computation Element (PCE) Architecture, RFC 4655, August 2006.**
- **H. Smit and T. Li, *IS-IS Extensions for Traffic Engineering*, RFC3784, June 2004**
- **J.P. Vasseur and J.L. Le-Roux, *Path Computation Element (PCE) Communication Protocol (PCEP)*, RFC 5440, March 2009.**
- ***L. Andersoon and G. Swallow, The Multiprotocol Label Switching (MPLS), Working Group Decision on MPLS Signaling Protocols, RFC3468, February 2003.***
- ***BFDWorking Group, http://ietf.org/html. charters/ bfd-charter.html***

- ***J. Le Roux, J.P. Vasseur, J. Boyle et al., Requirements for Inter-Area MPLS Traffic Engineering, RFC4105, June 2005.***

- ***R. Zhang and J.P. Vasseur, MPLS inter-AS traffic engineering requirements, RFC4216, November 2005***

- ***E. Rosen and Y. Rekhter, BGP/MPLS VPNs, RFC2547, March 1999.***

- ***Y. Rekhter and E. Rosen, Carrying Label Information in BGP-4, RFC3107, May 2001.***

- ***V. Alwayn, Advanced MPLS Design and Implementation, Cisco Press, 2001.***

- ***P. Pan, G. Swallow and A. Atlas, Fast Reroute Extensions to RSVP-TE for LSP Tunnels, RFC 4090, May 2005.***

- ***MPLS Traffic Engineering - Interarea Tunnels. (n.d.). Retrieved December 07, 2016, from http://www.cisco.com/c/en/us/td/docs/ios/mpls/configuration/guide/12_2sy/mp_12_2sy_book/mp_te_interarea_tun.html***