Technical report, IDE1104, February 2011

MPLS VPNs with DiffServ – A QoS Performance study

Master's Thesis in Computer Network Engineering

Azhar Shabbir Khan

Bilal Afzal

School of Information Science, Computer and Electrical Engineering
Halmstad University

MPLS VPNs with DiffServ – A QoS Performance Study

Master's Thesis in Computer Network Engineering

AZHAR SHABBIR KHAN

BILAL AFZAL

Supervisor:

URBAN BILSTRUP

School of Information Science, Computer and Electrical Engineering

Halmstad University

Box 823, S-301 18 Halmstad, Sweden

February 2011

Preface

First of all we would like to thank all mighty Allah for His countless blessings to complete our studies. After that we would like to thank Urban Bilstrup, Tony Larsson and Olga Torstensson for their kind support and helpful supervision throughout our thesis work that made us able to achieve our goal.

Finally we would thank to our parents and all family members whose support always encourages and empowers us throughout our studies.

Azhar Shabbir Khan & Bilal Afzal

Halmstad University, February 2011

Abstract

MPLS (Multiprotocol Label Switching) VPNs (Virtual private network) are new alternatives to private WANs (Wide area network). They are gaining popularity in industry day by day. Enterprise customers are moving to service providers that offer MPLS VPNs. The main reason for this shifting is the capability of MPLS VPN to provide built in security features and any-to-any connectivity. QoS (Quality of service) is the most important element for enterprise networks. Enterprise network have video, audio and data traffic over a single network infrastructure.

In this thesis we will investigate QoS parameters (e.g. delay, jitter and packet loss) over MPLS VPNs environment. It will help the service providers and enterprise network customers to maintain QoS for voice, video and data traffic over MPLS VPNs environment.

To achieve consistent end-to-end QoS, the service provider and the enterprise customer must work closely together and share the same policies to implement because service provider participates in customer routing in MPLS VPN environment. We will use the DiffServ (Differentiated services) QoS model over MPLS VPN network. We will use a six class model for service provider network and enterprise customer network to achieve end-to-end service quality.

In the last part we will make tests of end-to-end traffic delay, jitter and packet loss. We will compare the results of delay, jitter and packet loss with and without the DiffServ QoS model in an MPLS VPNs environment. It will be clear in the results that without using DiffServ QoS model delay, jitter and packet loss are increasing as the traffic increases on the network. With a DiffServ enabled network, the increase of traffic over network will not affect delay, jitter and packet loss and provide constant level of service quality.

Contents

Introduction

Every day new technologies are being developed. Enterprises use these new technologies to upgrade their network services and reduce cost. Enterprises use single network infrastructure for all real time and data traffic. In past years, real time traffic and data traffic were sent over different network infrastructures. Companies want to use all these services on a single network to reduce effort and cost. New network infrastructures are designed on the basis of these requirements. Voice and video are two different types of applications and require a service guarantee all over the network. If real time traffic is sent on the ordinary network then there will be longer delays compared to bearable delays. Service quality is required for real time traffic. Service quality over MPLS VPNs makes this technology very useful for enterprise networks. To achieve service quality for real time applications like video and audio, DiffServ QoS model can be used with MPLS VPN. MPLS is a fast packet forwarding technology [16]. MPLS VPN with DiffServ provides better service compared to Pure IP (Internet Protocol) based networks. It provides a service guarantee in terms of constant end-to-end delay, jitter and packet loss.

1.1 Motivation

MPLS VPNs is a new technology, which is rapidly replacing other WAN technologies. Enterprise networks are moving towards Internet for WAN connectivity. In past years, enterprise networks were using Frame Relay, ATM, T1 or E1 dedicated lines for WAN connectivity. To fulfill security requirements, enterprise networks were using Layer2 VPNs. Layer2 VPNs are not scalable, and provisioning of new sites is more complex compared to the provisioning of MPLS VPNs [16]. MPLS VPNs overcomes the limitations of WAN connectivity and layer2 VPNs. It provides Layer3 VPNs that covers the security over the network. Layer3 VPNs logically separates whole VPN network from others. Each VPN maintains their routing table and is not mixed up with other networks [3]. They also provide the benefit of less overhead compared to the other layer2 VPN technologies. Provisioning of new sites is very easy; it requires only connectivity to the service provider.

The DiffServ QoS model provides better control and administration of network traffic. It limits the different applications to use the network resources as per business needs. The outcomes of this thesis will help network engineers to make their traffic management decisions. It will provide better understanding as to how DiffServ QoS model and MPLS VPNs work together and how it is useful for enterprise network that run video, audio and data traffic over the same network infrastructure.

1.2 Problem Studied

MPLS VPNs provide new WAN connectivity options for enterprise networks. DiffServ QoS model is used to manage network traffic in better way and it is helpful to take the optimum results from available network resources. Service quality is required to facilitate the video and audio traffic over the congested network. MPLS VPNs combines WAN connectivity and VPN options but service quality is still required to better facilitate video and voice traffic.

Real time traffic is affected by network delays, jitter and packet loss. To achieve a consistent level of service quality for real time traffic, it is required for MPLS VPN networks to use any QoS model. In this thesis we will use DiffServ QoS model over MPLS VPNs and analyze how they work together and what the effects of this will be. Enterprise network customers demand approximate constant value of delay, jitter and packet loss to better facilitate the voice and video traffic applications over the MPLS VPNs. The DiffServ QoS model is chosen because it is very flexible and scalable, and works on a hop by hop basis instead of source to destination flow, as in the integrated service (IntServ) framework [3].

1.3 Goals

In this thesis we will analyze the effect of using DiffServ QoS model over MPLS VPN. We will study MPLS VPN network behavior with and without using DiffServ QoS model. We will then make our analysis on the basis of delay, jitter and packet loss in different provisioning states of network and on different traffic loads.

1.4 Structure of thesis

Chapter 1 is the introduction of our work and explains application area, motivation, problem studied and our goal. Chapter 2 will explain MPLS technology, MPLS structure and MPLS operations Chapter 3 focuses on MPLS VPNs. It defines virtual private network, VPN models, MPLS VPN Model, MPLS VPN architecture, MPLS VPNs benefits and MPLS VPN support to QoS. Chapter 4 defines the QoS, QoS issues and QoS models. Chapter 5 is about the DiffServ QoS model configurations, in which classification, marking and per-hop behavior is defined. Chapter 6 is about the practical work of our thesis work and it includes the details of a working scenario and configurations of MPLS VPNs with DiffServ QoS model. Chapter 7 will define the testing results and compare the results of different scenarios.

2 Multi Protocol Label Switching

2.1 Overview

Multi protocol label switching (MPLS) is a technology developed by the IETF (Internet engraining task force) to overcome the problems of traditional IP routing and to make routing fast, manageable and able to carry heavy traffic, and accept new routing architectures. MPLS is a modern technique for forwarding network data. In a MPLS network packets are assigned labels and the labels are used to make forwarding decisions without IP lookups at each node. It is called multi protocol because it supports any layer 3 network protocols. MPLS work between layer 2 and layer 3 which is called layer 2.5 technologies. MPLS provides the scalability for the Virtual private networks (VPNs) and support for end-to-end quality of service (QoS) [1].

2.2 Benefits of MPLS

1 **VPNs:**
The backbone network can be created by using layer 3 VPNs by the service providers [3].

2 **Traffic engineering**
MPLS provide the single or multiple network traffic paths explicitly. It also provides the ability to set the performance characteristics for a class of traffic [3].

3 **Quality of service**
Provides guaranteed quality of service (QoS) for the for their VPN customers for multiple classes for service [3].

2.3 MPLS Working:

Figure 1 shows the MPLS routing process in larger networks. There are two types of routers, edge routers and core routers. The routing decisions are made only at the edge routers and the core routers forward packets based on the labels. These two functions provide fast forwarding method of packets.

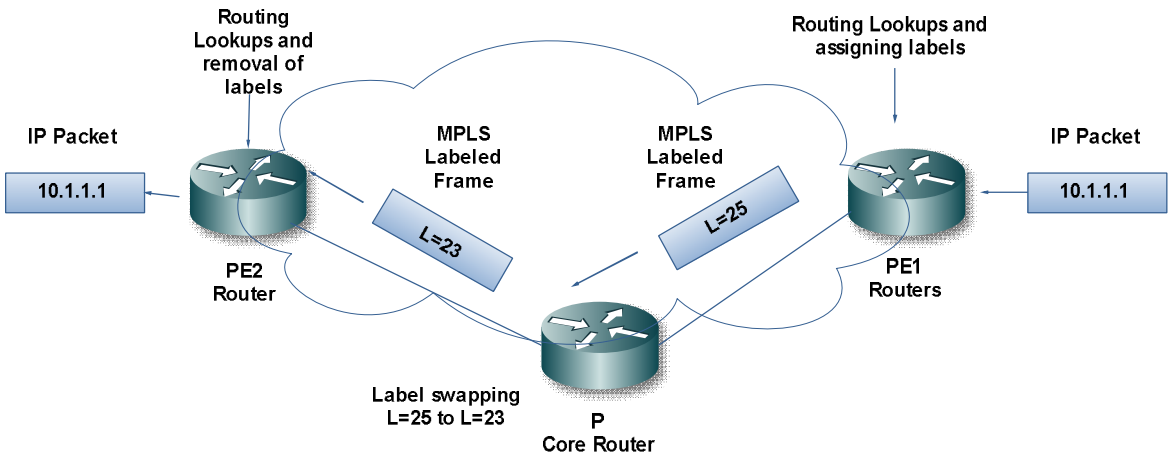


Figure 1: MPLS working

In the Figure 1, PE1 and PE2 routers are edge routers and P is a core router. The IP packet with IP address 10.1.1.1 goes to the PE1 (MPLS enabled router), the PE1 router perform routing lookups and attaches a label, 25, and sends it to the core router P. The core router then swaps the label with new label, 23, and sends the packet to PE2 edge router. The PE2 router performs routing lookups and removes the label and sends it to the destination as a simple IP packet. The packet will then go through the path called the Label Switched Path (LSP).

2.4 MPLS Header

The MPLS header consists of 32 bits. The first 20 bits are used for the actual label and these bits are called label bits. The three bits are called experimental bits, these bits are used by Cisco to define a class of service (CoS). MPLS enabled routers might need to insert multiple labels to send packets through the MPLS network. To determine which label is the last label in the packet, a bottom of the stack (BoS) bit is used, if the bit is 1 it means that it is the last label. The last 8 bits are used to time to live (TTL) they have the same function as the usual IP header.

Figure 2: MPLS header

2.5 Label Stacking

MPLS routers sometime need more than one label in front of the IP packet header to route that packet through the MPLS network. This is done by packing the labels into a stack. The first label in the stack is called the top label, and the last label is called the bottom label. In between, you can have any number of labels. Figure 3 shows you the structure of the label stack [2].

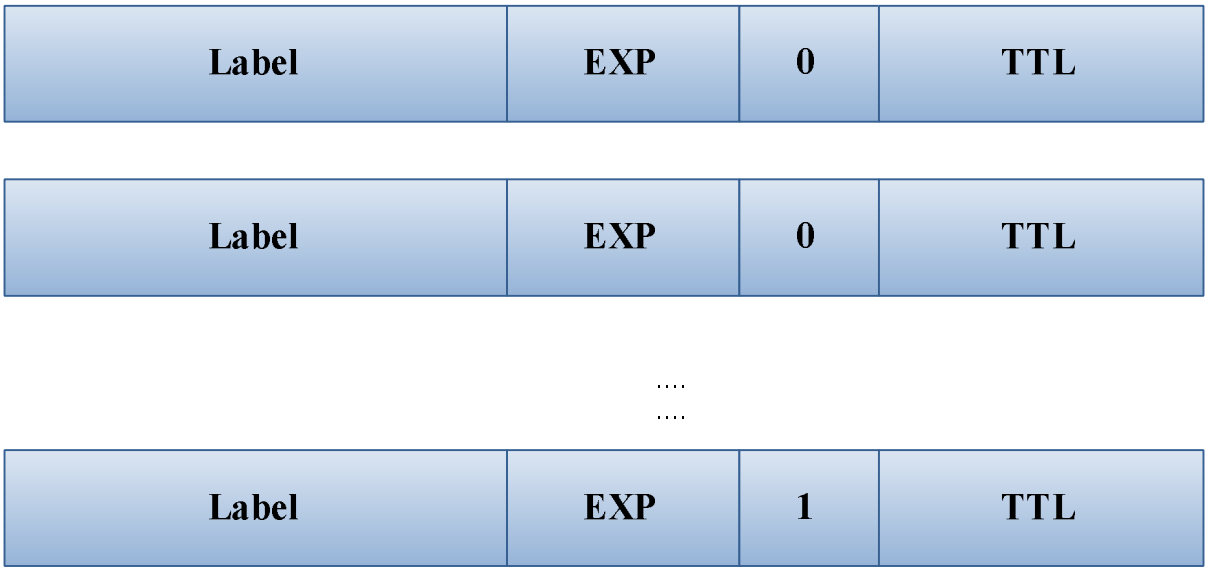


Figure 3: MPLS Label Stack

2.6 MPLS Architecture

2.6.1 Control Plane

The control plane is responsible for the routing information exchanges and the label information exchanges with the adjacent routers. Link state routing protocols advertise routing information among the routers that are not necessarily adjacent, where-as label binding information distribution is limited to adjacent routers. [3]. Control plane consist of two types of protocols. The routing protocols (e.g. OSPF, BGP, IS-IS, RIP, EIGRP) and label exchange information protocols. Including the MPLS LDP label distribution protocol and BGP, this is used by MPLS VPN.

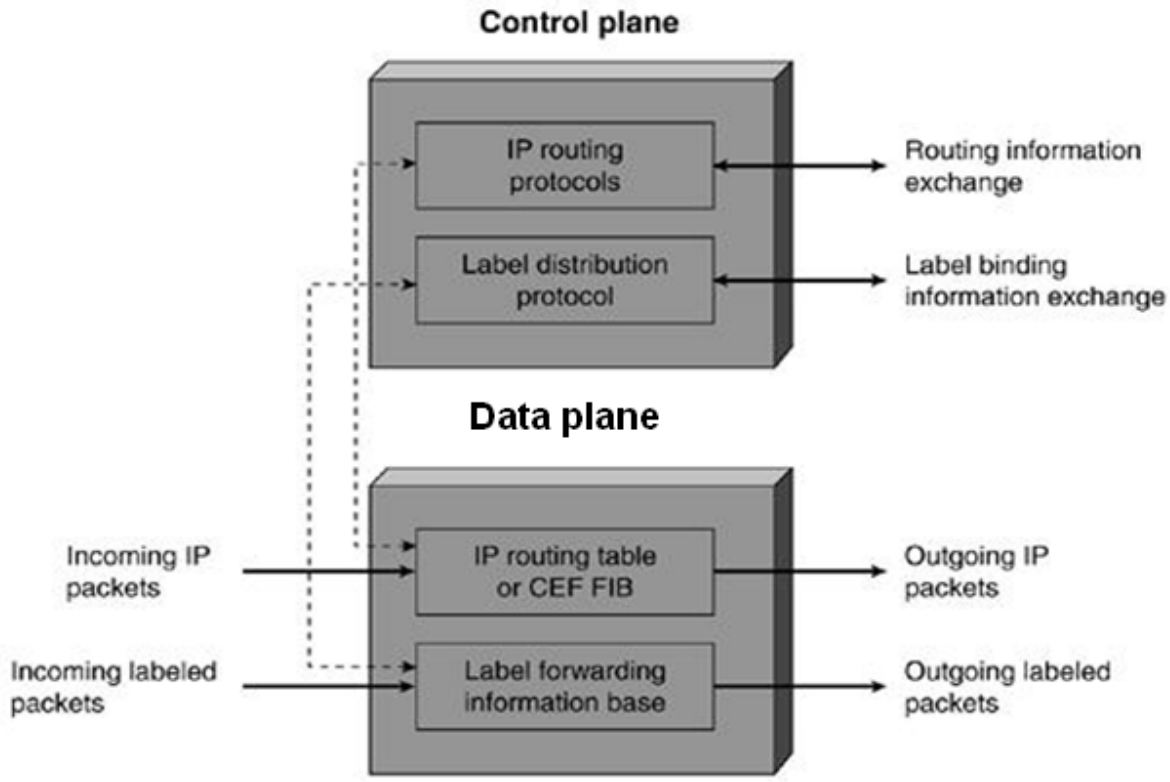


Figure 4: MPLS Architecture [3]

2.6.2 Data Plane

The MPLS data plane has a simple forwarding engine, based on the information attached with labels. There are two tables on each MPLS router, LIB and LFIB. The data plane uses a label forwarding information base (LFIB) maintained by the MPLS enabled router to forward labeled packets. The LIB table contains all the local labels assigned by the local routers and mapping of the labels that it receives from the adjacent MPLS routers. The LFIB uses a subset of the labels contained in the LIB for actual packet forwarding [3].The MPLS enabled routers use information in LFIB and label value to make forwarding decisions [3].

2.7 Label Switch Routers

A label switch router (LSR) [1] [2] is a router that supports MPLS. These routers have ability to understand the MPLS labels and they can receive and transmit labeled packets. There are three kinds of LSR's.

2.7.1 Ingress LSR

Ingress LSR's [2] receive an unlabeled packet, insert a label in front of packet and send it to a data link.

2.7.2 Egress LSR

Egresses LSR's [2] receive a labeled packet and remove the label and send it to data link. Ingress and egress routers are edge routers.

2.7.3 Intermediate LSR

Intermediate LSR's receive an incoming labeled packet, perform an operation on it, switch the packet, and send the packet on the correct data link. [2]

2.8 Label Switch Path

Label switch path (LSP) [2] is the path that a packet passes through from ingress LSR to the intermediate LSR and then the egress LSR. Figure 5 bellow gives the graphical view of an LSP.

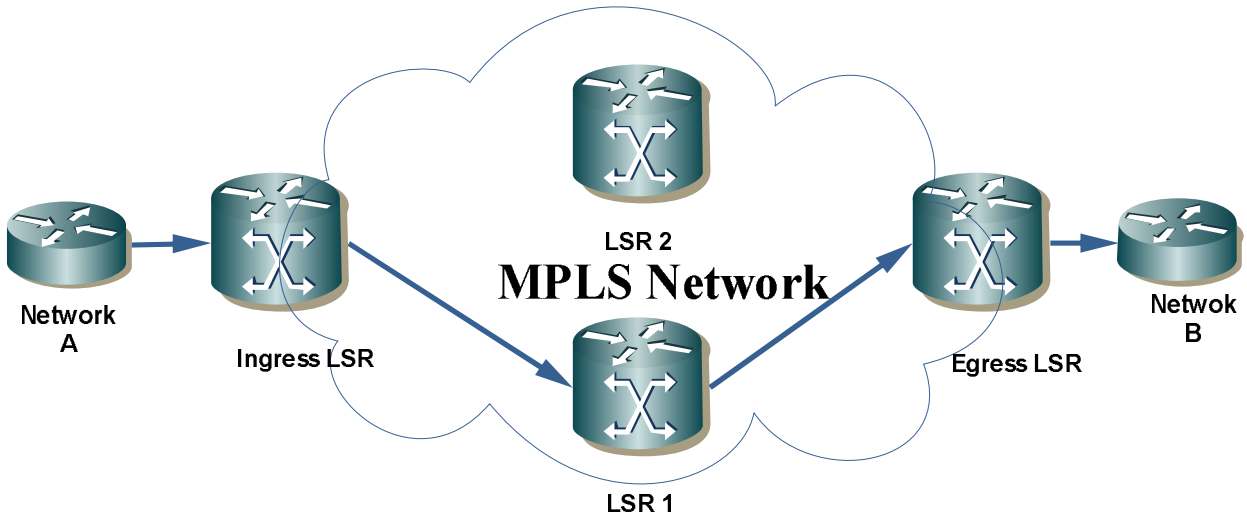


Figure 5: Label Switch Path

The packet from network "A" takes the path from ingress router via LSR1 router and the egress router to reach the destination Network B.

2.9 Label Distribution

The packet in the MPLS network [1][2] that has to pass through the network is forwarded through the label switch path (LSP) tunnel. When the packet reach the MPLS network then the Ingress router receives the packet and puts MPLS label in the packet and sends it to the next hop according to the destination address in the packet. There can be many LSRs between Ingress and Egress routers, so when packet reaches a LSR it swaps the labels and sends it to the next LSR. When the packet reaches the Egress router, it strips off all the labels and sends it to outgoing. All the LSRs have interior gateway routing (IGP) e.g. OSPF, EIGRP, RIP are running throughout the network. To accomplish this task, the adjacent LSRs must agree on a label that will be used as prefix of IGP and each LSR must know which label should be swapped for incoming and outgoing packets. This shows that we need a mechanism to tell routers which label will be use when forwarding a packet. Each pair of router labels is local and they do not have any global meaning across the network. There must be some communication between the two adjacent routers to exchange label information. Otherwise the routers do not know which incoming label need to match with which outgoing label. For this purpose label distribution protocol is needed.

There are two ways to distribute labels

- Piggyback the label on the existing routing protocol
- A separate routing protocol for label distribution.

2.9.1 Piggyback the label on the existing routing protocol

There is no need for a new protocol on LSRs but they have to extend the existing routing protocol to carry the labels and it is not an easy task. For distance vector routing protocols its implementation is easy because each router originates a prefix from its routing table. The router then just binds a label to that prefix.

Link state routing protocols work different from distance vector routing protocol. In link state routing protocol, each router generates link state updates and forwards the unchanged link state information to all routers inside one autonomous area. A problem with MPLS is that it needs to distribute labels for each Interior gateway protocol prefix even to the routers which do not generate prefixes. In order to do this the link state routing protocol needs to be enhanced. It is a difficult task so for link state routing protocols a separate protocol is preferred for the distribution of labels. There is one routing protocol in MPLS VPN that can distribute labels and carry prefixes at the same time which is called border gateway routing protocol (BGP). In MPLS VPN networks BGP is a routing protocol that can carry prefixes and distribute labels at the same time [2].

2.9.2 Separate Routing Protocol for Label Distribution.

Running a separate protocol for label distribution is a 2nd method. The advantage of this method is that routing and label distribution protocols are separate and the disadvantage of this is that each LSR runs an additional protocol.

The label distribution protocols are.

- Tag Distribution Protocol (TDP)

- Label Distribution Protocol (LDP)
- Resource Reservation Protocol (RSVP)

TDP is a Cisco proprietary protocol developed and implemented by Cisco. LDP is designed and developed by IETF. TDP and LDP operate in a very similar way but LDP has more functionality than TDP and LDP has replaced TDP very quickly. The recourse reservation protocol (RSVP) is only used for MPLS traffic engraining (TE).

2.9.3 Label Distribution Protocol (LDP)

In a MPLS networks each LSR assigns a label to every IP prefix in its routing table and it is local binding. Then LSR distribute these bindings to its neighboring routers. For neighboring routers these bindings are remote bindings. Remote and local bindings are stored in a table name local information base (LIB) [2].

2.9.4 Label forwarding information base (LFIB)

This table is used to forward labeled packets. The LFIB table is populated with outgoing and incoming labels of LSPs. The information in LFIB about the labels is used to forward packets in the MPLS network. [2]

2.9.5 MPLS Payload

Only the egress routers are aware of the payload because LSRs in LSP need information only about labels to make forwarding decisions. All the labels are removed by the egress router in MPLS domain.

2.9.6 Forwarding equivalence class

The packets have the same characteristics that are considered in the same class. For those packets the same label is used and the class is known as a forwarding equivalence class. [2]. A specific LSP can be used for the multiple forwarding equivalence classes.

2.10 Different MPLS Modes

When labels are distributed among different LSRs, the MPLS uses the following three types of modes.

- Label distribution mode
- Label retention mode
- LSP control mode

Label distribution mode

For label distribution MPLS use two modes

1. Downstream on demand label distribution (DoD)
2. Unsolicited downstream label distribution (UD)

In downstream on demand label distribution mode each LSR request for the label is binding to its downstream LSR.

In unsolicited downstream label distribution mode neighbor LSRs distributes labels amongst one another. In UD more than one binding is shown and in DoD only one binding is shown [2].

2.10.2 Label retention mode:

It has two types of modes

1. Liberal label retention mode (LLR)
2. Conservative label retention mode (CLR)

In LLR mode all received bindings from LSRs are kept in the LIB. Triggers updates are sent to update the LIB when topology changes occur. [2]. If a down LSR replaced with new one the information is updated in LFIB quickly. [1][2]. In CLR mode all bindings are not stored in the LIB, only the remote bindings associated to the neighbor's next hope are stored in the LIB [1] [2].

2.10.3 LSP control mode:

There are two modes:

- Independent LSP Control mode
- Ordered LSP Control mode

The LSR can create a local binding for a forwarding equivalence classes (FEC) independently from the other LSRs. This is called Independent LSP Control mode. In this control mode, each LSR creates a local binding for a particular FEC as soon as it recognizes the FEC. Usually, this means that the prefix for the FEC is in its routing table.

In independent LSP control mode the LSR can create local bindings independently for FEC from other LSRs. In Ordered LSP control mode, LSR only creates a local binding for a FEC if it recognizes that it is the egress LSR for the FEC or if the LSR has received a label binding from the next hop for this FEC [2].

3 Multiprotocol Label Switching and Virtual Private Networks

3.1 Virtual Private Network

A virtual private network is a network that connects private networks over the public network. VPNs provide connectivity on OSI layer 2 and layer 3. Service providers use VPNs to interconnect different sites that belong to same corporation. A requirement of a corporation's private network is that all their customer sites VPNs remain separate from the other corporation VPNs [2].

At the IP layer, VPN models might require that different VPNs are required to connect with one another and also provide connectivity to the internet. MPLS VPN provides this functionality. Service providers use MPLS in their backbone network which supplies a decoupling of the forwarding (date) plane and the control plane that IP does not [2].

3.2 VPN Models

There are two types of VPN models.

- Overlay VPN
- Peer to peer VPN

In overlay VPNs service provider provide point to point virtual links and in peer to peer VPN model, routers participate in customer routing.

3.2.1 Overlay VPN [3]

- Layer 2 VPN
    1. X.25
    2. Frame relay
    3. ATM
- Layer 3 VPN
    1. GRE
    2. IPSec

3.2.2 Peer to peer VPN [3]

- ACLs
- Split Routing
- MPLS VPN

3.3 MPLS VPNs Benefits

MPLS VPN is a new and robust technology which reduces the complexity of network. It provides a complete communication solution for enterprise network. Multiple applications data can be sent over it. There is no need to manage the individual connection between different site offices. Enterprise customer only needs to link up with service provider. The service provider is responsible for customer routing and multiple sites connectivity.

MPLS VPN reduces the cost of network operations. There is no need to hire highly technical manpower to manage the network operations because the service provider is responsible for all the sites' connectivity. There is no need to deploy the equipment for each site office. MPLS VPN also supports the encryption of data. Customer can use different encryptions (public key and private key) to protect the data.

3.4 MPLS VPN schematic overview Model

It is important to know terminologies that are related to MPLS VPN. Figure 7 gives an overview of the structure of the MPLS VPN network. IPSs provide the infrastructure to connect the different customer sites.

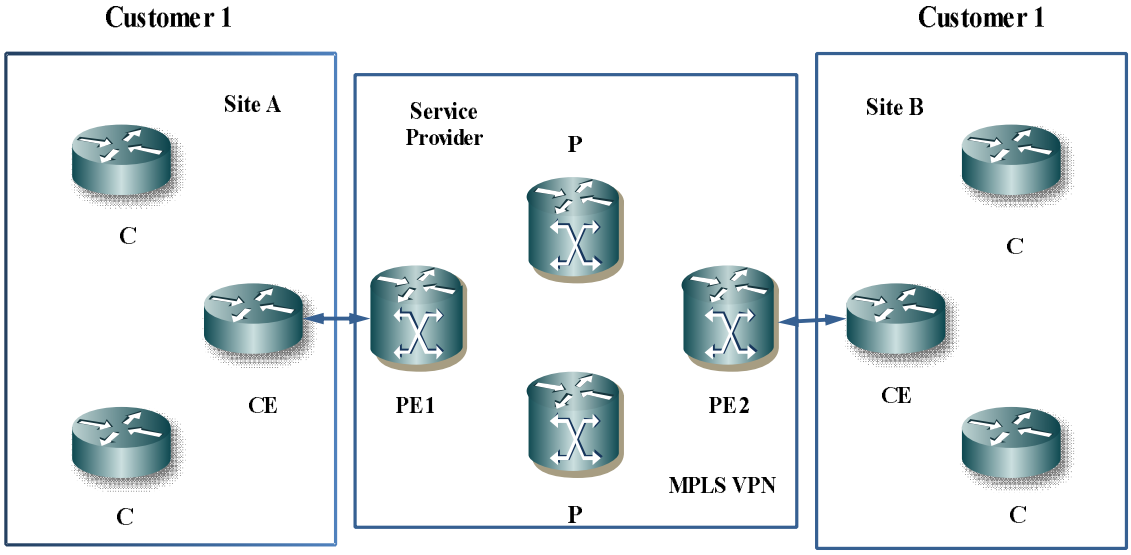


Figure 6: MPLS VPN Model

At the service provider network there are two types of routers, Provider Edge (PE) and Provider (P) routers. The PE router is directly connected with the customer edge (CE) routers at layer 3. The P router doesn't have any direct connection with the CE routers. Both PE and P routers run MPLS so they can distribute MPLS labels.

Figure 6 CE routers are directly connected with the PE1 and PE2 router at layer 3. A Customer Router (C) has no direct connection with PE1&PE2 routers of provider's network. There is no need to run MPLS on a CE router because CE and PE1&PE2 routers interact by using some routing protocol or static router at layer 3. There is no peering between CE routers on different site as in overlay VPN. The concept of peer to peer model derived from CE and PE1&PE2 routers peer at layer 3.

3.5 MPLS VPN model

The service providers allow their customers to use any kind of IP address like a registered IP address or a private IP address (RFC 1918). As customer can use any kind of IP address, there is a possibility that different customers have same IP addresses that are connected to same service provider (overlapping of IP address). If the packets are forward as

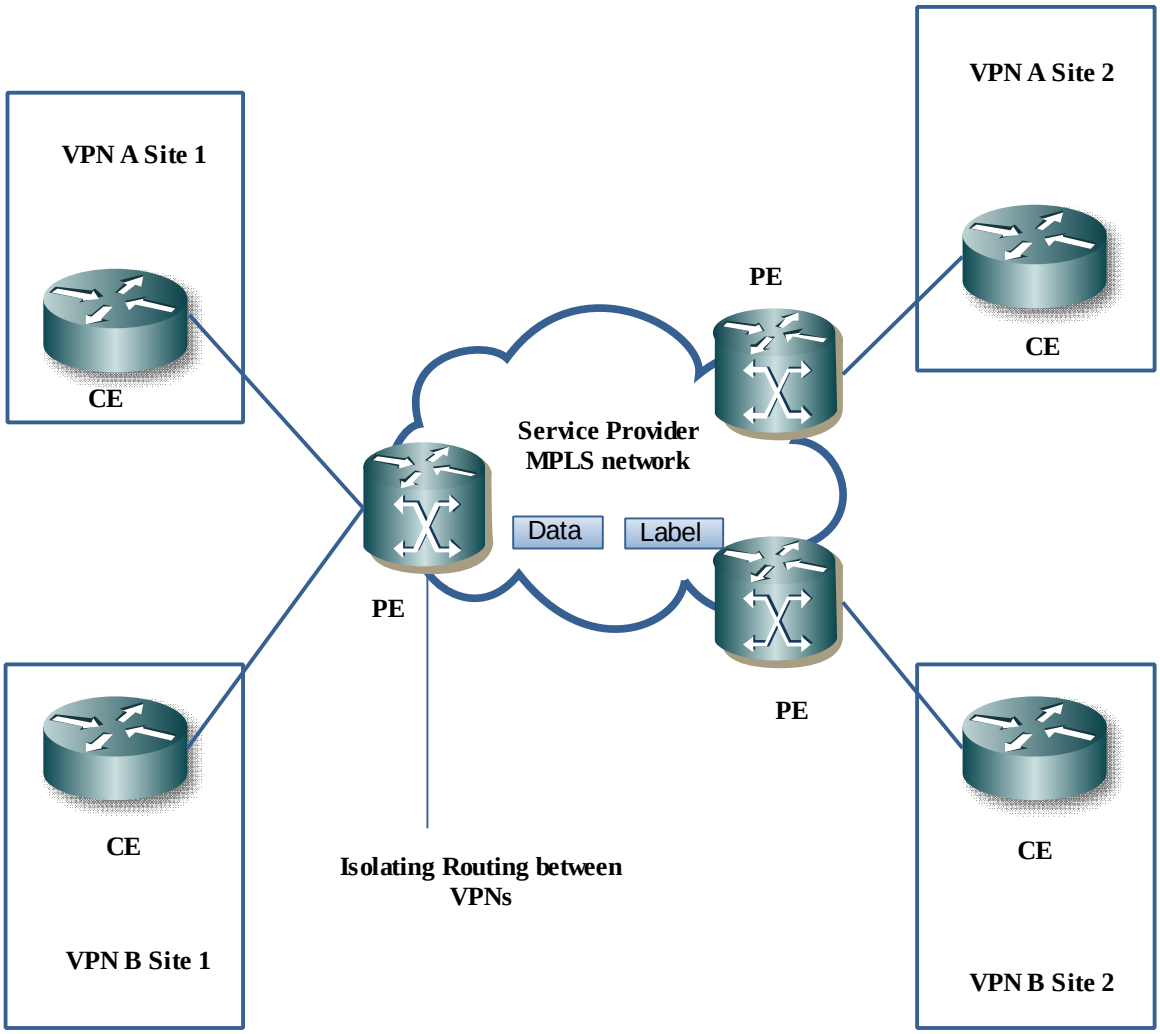


Figure 7: MPLS VPN Model

simple IP packets then for P routers it is difficult to find out the destination. If overlapping and private addressing are not allowed, then every router in providers network lookup in destination address and forward the packets. This means that all P and PE routers must have a complete routing table for every customer. For this large routing table BGP is used to carry these routes so all service provider routers use iBGP as routing protocol inside their network [2].

The solution would be that the P routers completely remain of about VPNs and they will have no burden of having routing information for VPN routers. This can be achieved by using MPLS. The IP packets that are coming from the customer side are label at the service provider's network at PE routers to achieve a private VPN for each customer. There is no role for P routers in the customer routing tables and BGP have no more need at the core network.

Only PE routers have knowledge of VPN routes. As shown in Figure 7, the VPN information is only on PE routers and the MPLS VPN network [2].

3.6 MPLS VPN Architecture

There are some basic building blocks for the MPLS VPN at PE routers. These are given below. [2]

1. Virtual Routing Forwarding (VRF)
2. Route Targets (RT)
3. Route Distinguisher (RD)

3.6.1 Virtual Routing Forwarding:

The combination of the VPN IP routing table and the associated VPN IP forwarding table is called the VPN routing and forwarding instance (VRF) [4]. VRF is used to make the MPLS VPN networks private. The VRF makes sure that the routing information is kept separate from different customers and that the backbone of the MPLS network makes sure that the packet forwarding is based on label information and not on the information in the IP header.

VRF is the routing and forwarding instance of VPN. It is a combination of the VPN routing table, the VRF CEF table and IP routing protocols on PE router. A PE router contains a VRF instance for each VPN that is attached to it. As shown in figure 8 the global IP routing table with the VRF routing table of both sites are attached with the PE router.

On PE routers each VPN has its own separate routing table and this routing table is called the VRF routing table. A PE router interface that is towards the CE router only has one VRF, so that all IP packets coming to that interface will be consider as they are belonging to that VRF. It is because there is a separate routing table per VPN. There is a separate CEF table per VPN to forward these packets on the PE router which is called the VRF CEF table. As with the global routing table and the global CEF table, the VRF CEF table is derived from the VRF routing table [2] [4].

An interface can only assign to one VRF, but several interfaces can be assigned to the same VRF. The PE router then creates a VRF and CEF table. The VRF routing table is similar like regular routing table and it is used for a set of VPN sites and is separated from all other routing tables.

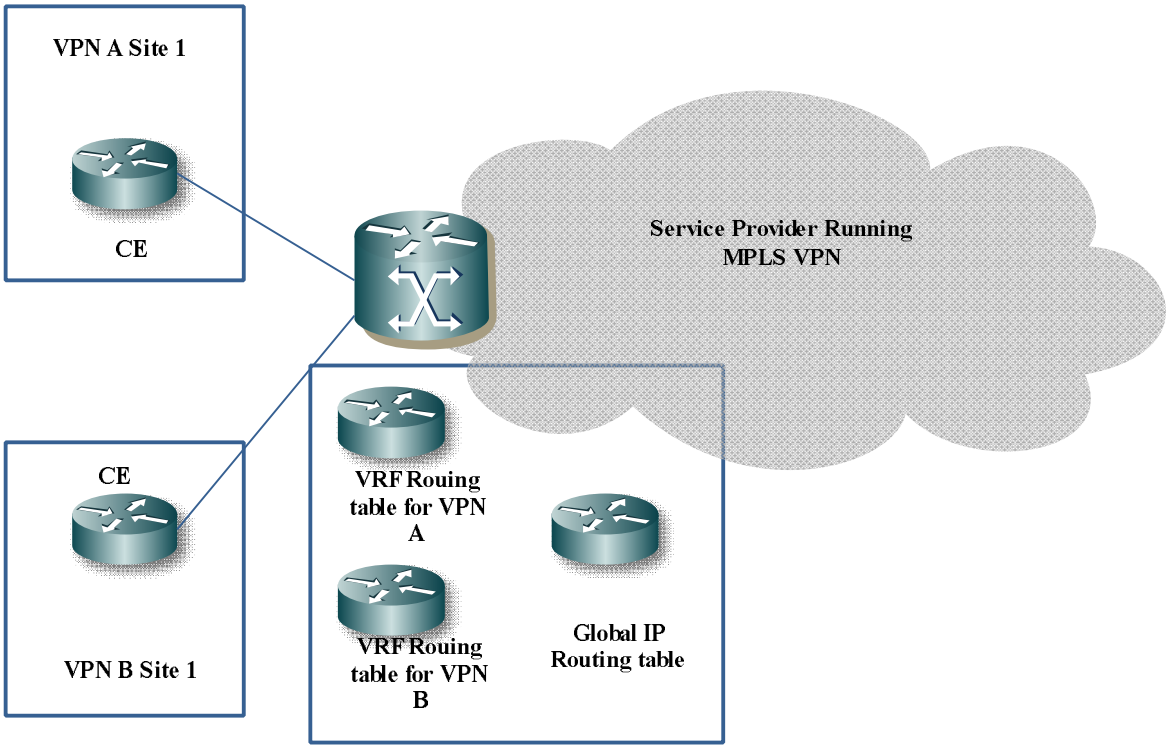


Figure 8: VRF on PE Router

3.6.2 Router Distinguisher

The VPN uses Multiprotocol BGP (MPBGP) to propagate its prefixes in MPLS VPN networks. The IPv4 prefixes must be unique when they cross the service providers' network. If there is overlapping in customers IP addressing then routing is a problem. To overcome this problem the concept of route distinguishers (RD) is used to make the IPv4 prefix unique. The combination of IPv4 and RD is called VPNv4 prefix. MP BGP is used to carry these VPNv4 prefixed between the PE routers. [2][4]

The RD is 64 bit field. One RD must be assigned to each VRF at the PE router. The 64 bit value can have two formats IP Address:nn or ASN:nn where nn is number and ASN is the autonomous system number. Mostly service providers use ASN:nn. ASN is assigned by IANA to service provider and "nn" is uniquely assigns to VRF by the service provider. RD and IPv4 prefix provides a VPNv4 prefix and it is 96 bit long address and subnet mask is 32 bit long. [2]

If the IP address is 10.10.10.1/24 and the RD is 2:2 then VPNv4 prefix will become 2:2:10.10.10.1/24. [2]

3.6.3 Route Target

A route distinguisher works fine but the problem is that they can only communicate with one VPN. To overcome the problem router targets (RT) are introduced. RTs are able to communicate between complex VPN topologies.

RT is attached as an additional attribute to VPNv4 BGP routes to indicate the VPN membership. RT indicates which route should be imported from the Multiprotocol BGP into

the VRF. The RT that is attached with the route are called the export route and configured separately for each virtual routing table in a PE router. In MPLS VPN architecture the RTs at PE routers are attached with the customer route when it is converted from IPV4 to VPNv4 route [2].

When a PE router propagates the VPNv4 address to other PE routers, those routers have to select the best router to import into their virtual routing table. This selection is based on import RT. At the PE routers each virtual routing table have a number of import RTs that indentify the set of VPNs that the virtual routing table is accepting routes from [5].

4 Quality of Service (QoS)

4.1 Overview

QoS is the mechanism of the network to provide different service level to a different traffic type as business need. [6]

Service providers offer their network service with quality. They define a Service-Level Agreements (SLA). SLA provides the details of all QoS parameters. It defines the parameters such as end-to-end delay, end-to-end jitter, packet loss. QoS is not single device functionality and it is an end to end mechanism. It provides the intelligence to network devices to treat the different application's traffic as their defined service level by SLA. QoS combines different technologies together such as classification, marking, scheduling, queuing, bandwidth allocation, and prioritization that are commonly used to provide a scalable end to end service [3].

QoS is a generic term. It provides the different level of treatment to the different types of traffic or applications that flows over network. Quality of service is required to provide the well management of network resources that makes the sophisticated usage of resources and gives comfort to network user. Business networks are widely expended with different types of applications. These applications have different network requirements. It needs to lead for different administrative policies that control applications as per their requirements individually. QoS within a network is essential to meet the requirements of today's converged networks. QoS provides the different levels of service for business critical application and delay-sensitive applications.

QoS is to manages the following network elements.

1. Bandwidth: Maximum amount of date that can be carried.
2. Delay: The time to send data from source to destination.
3. Jitter: Variation in delay.
4. Reliability: Packet loss.

4.2 Issues to address for Quality of Service

Converged networks support different types of application such as voice, video, critical data, browsing, and network management. Different applications have different level of sensitivities and different requirement. These applications run on same infrastructure so it is a challenge to fulfill the requirement of application as per requirements. Some applications are delay sensitive, some application requires more bandwidth, some applications require constant amount of bandwidth and some applications require less packet loss (reliability). For example voice over IP (VOIP) applications are delay sensitive and run smoothly on maximum 150ms to 200ms end-to-end delay. On the other hand file transfer protocol (ftp) is not delay sensitive and jitter also does not effect on it. Some applications are TCP-based. If TCP segment is dropped, source retransmits that segment after the time out. TCP based applications can tolerate on packet drop. Some applications use UDP and have no acknowledgement mechanism. These applications cannot tolerate on packet drop such as VOIP, video over IP, online gaming. Converged network run different applications simultaneously. It is necessary to handle all applications individually and use some mechanisms that handle applications properly according to its nature.

There are four major challenges in converged campus network

1. Bandwidth
2. Delay
3. Jitter

4. Packet loss

4.2.1 Bandwidth

The amount of data that can be transmitted over link is bandwidth. On the network IP Packets travel though the best route. Maximum bandwidth of the route is equal to smallest value of bandwidth on route. [6]

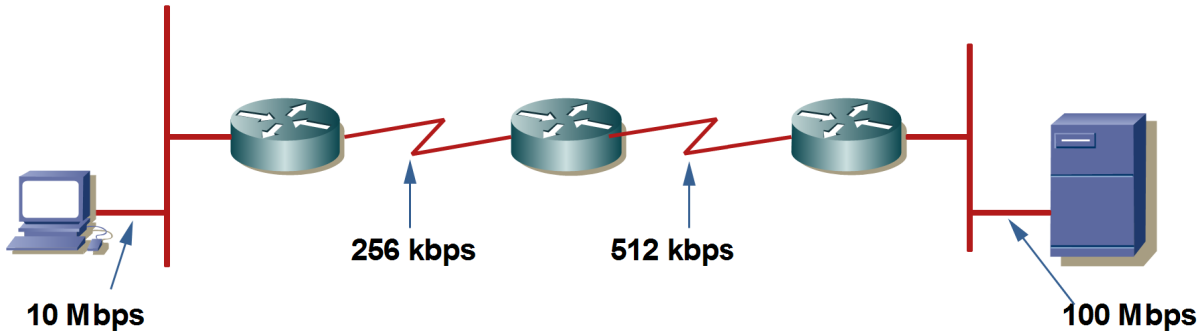


Figure 9: Bandwidth

Figure 9 shows the path of packet that has different bandwidth on links. The Bandwidth of the path will be 256kbps because it was minimum bandwidth on the path. The available bandwidth is the path bandwidth divided by number of traffic flows. [6] Due to the low bandwidth users experience delay, jitter and packet loss in the communication. This problem can be overcome by multiple ways.

- Increase link bandwidth:
  This is effective but costly
- Classify and mark traffic and apply queuing:
  Forward important packet first
- Use Compression technique:
  Layer2 payload compression, TCP header compression, and compressed RTP (cRTP) are some examples [6]. Usage of hardware compression is preferable over software based compression because compressions are CPU intensive and create delay.

4.2.2 Delay

End to end delay is the total time that a packet takes from source to destination. [6] End-to-end delay is sum of all the following delays.

1. Processing delay
2. Queuing delay
3. Serialization delay
4. Propagation delay

4.2.3 Jitter

Variation in delay is jitter. Packets for the same destination may not arrive at same rate. Campus network run different applications simultaneously. Jitter can occur due to different traffic load on different timings. For voice and video it is necessary to receive the packets at same sequence to achieve good quality. [6].

4.2.4 Packet loss

Packet loss occurs due to the low buffer space. When the buffers space of the interface full then packets are dropped. Packet loss creates extended delays and jitter. Packet loss can be controlled by applying some techniques such as Tail Drop, Random Early Detection, Weighted Random Early Detection and Traffic Shaping and Policing. [3]

4.3 QoS Models

There are three QoS models that are

1. Best-Effort Model
2. Integrated Services (IntServ) Model
3. Differentiated Services (DiffServ) Model

4.3.1 Best-Effort Model

Actually when we talk about Best-Effort model its mean no QoS is configured. In this model all the traffic is treated in the same manner and all are equally important. No classification and no differentiation in different applications. Best example of Best-Effort model is the normal post service. Best-Effort model provide scalability and Ease to handle but it has lack of service guarantee and lack of service differentiation.

4.3.2 Integrated Service Model (IntServ)

Integrated Service (IntServ) model was the first model that was developed to achieve end-to-end QoS. It was developed to fulfill the requirement for real time applications. Basic idea was to reserve the network resources for applications by guaranteeing bandwidth, delay and packet loss. Reservation of network resources to provide the service level resource reservation protocol (RSVP) is used. [6] It provides the signaling and reserves end-to-end network resources for the application. It is also called the hard QoS model. If it cannot reserve the recourses as per the policy it refuses to let the application operate. It is a virtual circuit and flow based model. [8]

4.3.3 Differentiated Service Model (DiffServ)

Differentiated Service Model comes after IntServ QoS model. It overcomes the limitation of the IntServ model. DiffServ is not a guaranteed QoS model but this model is more scalable. [6] It does not require end to end resource reservation, no need for signaling and no need to maintain per flow status. It is also called the “Soft QoS” model. IntServ model guarantees for the end-to-end resource reservation before application take start. It uses the RSVP for signaling and end-to-end resource reservation. DiffServ does not use the signaling protocols. It uses the per-hop-behavior (PHB). Each node in the network provides the specific level of service for each traffic class. PHB does not required end-to-end resource reservation while the decision is being made at each hop and provides the service level to the traffic class. [7]

4.4 Differences and limitations of IntServ and DiffServ

IntServ and DiffServ are two mechanisms to achieve QoS. They have different architectures. There are differences in their structure and configurations.

IntServ model was the first attempt to achieve QoS. It is based on per flow operations. It uses admission control to provide the guaranteed QoS for specific flow. It reserves bandwidth throughout the path and then allows the application to start if there is availability

of resources. It uses RSVP for reservation of resources. When RSVP reserves the bandwidth, that bandwidth cannot be used by other transmission. Each flow is isolated from one another. It works like a private leased lines and it provides the guaranteed QoS.

DiffServ model has different mechanism to provide QoS. It does not provide a guaranteed QoS but it is more flexible, it is like statistical multiplexing. It does not reserve the resource for each flow. Traffic treatment decisions are being made at each hop. It does not use any signaling protocol it uses per hop behavior to provide QoS. All the devices on network are preprogrammed to provide QoS for specific class of traffic. It is more flexible and scalable because it does not reserve end-to-end bandwidth for specific flow.

4.5 MPLS VPN QoS

On a per-VPN basis there are various MPLS QoS class of service (CoS) should be available. Real time applications should have different CoSs in the VPN real time applications such as VoIP should have a preferential CoS as compared file transfer or mail. QoS in context of VPN can be described in two models.

- Pipe model
- Hose model

4.4.1 Pipe Model

Service providers provide a certain QoS guarantee to the VPN customers for the traffic flows between the two CE routers within the VPN [3].

The pipe model can be represented as a pipe between two CE routers. The traffic that enters this pipe gets some QoS guarantees such as minimum bandwidth between two CE routers. PE routers are used to specify the requested traffic flows that are permitted to use this pipe. The MPLS QoS pipe model and the QoS model are similar so that VPN customers are familiar with Frame Relay and ATM. Frame Relay and ATM are bidirectional whereas a pipe model is unidirectional. The unidirectional nature of the pipe model allows traffic pattern irregularity which allows different traffic rates in each direction between the CE routers. The pipe model resembles the IntServ model for QoS and can provide hard guarantees [3].

4.4.2 Hose Model

With the hose model the service provider provides guarantees for the traffic that is sent and received by the particular CE router to other CE routers in the same VPN. In an MPLS VPN with QoS environment it is easy to configure hose model by the customer because the customer does not have to perform the capacity planning and traffic analysis and traffic distribution specifications between the different CE routers [3].

Two parameters called Ingress committed rate (ICR) and Egress Committed rate (ECR) are used in the hose model. ICR is the traffic rate in a VPN at which a CE receives traffic from the particular CE router, and in ECR the traffic rate in a VPN at which CE can send traffic to a particular CE router. The values of ICR and ECR are independent and do not need to be same [3].

DiffServ and hose models are similar in sense that they both support multiple CoSs. In the hose model CoSs are supported by using the DiffServ mechanisms.

Ingress PE routers determine which traffic receives a particular CoS. It depends on the IP precedence, IP source and destination, incoming interface, and TCP port numbers. The ingress PE routers can also police incoming traffic and mark packets that are out-of-rate, based on the service level agreement (SLA) agreed with the customer. These packets can be marked differently and dropped in case of congestion [3].

5 DiffServ over MPLS VPN

To achieve service quality in MPLS VPN environment we choose the hose model or DiffServ QoS model because it is widely used in industry due to its scalability as described in previous chapter 4. To use the DiffServ QoS model first step is classification. It is to classify the traffic into different classes. After classification each class is marked, this process is called marking. After marking, business policy for each class is configured as per service level agreement.

5.1 Classification

Classification is the process of dividing the traffic into different category. Each category is called a traffic class. Classification is the most fundamental part to achieve the QoS using DiffServ model. After making traffic classes, traffic becomes ready for further handling to achieve QoS. Classification is a processor intensive process but is happens once at customer edge router normally. The total effect of classification process does not make a major impact to end-to-end delay. Classification can be done by;

1. Incoming interface
2. IP precedence
3. Differentiated service code point (DSCP)
4. Source or destination IP address
5. Application
6. Five-Tuple (source and destination IP address, IP protocol number, TCP/UDP source and destination port numbers.). [15]

5.2 Marking

In simple wording marking is coloring the packet so that they are recognized. Marking is to place a value in differentiated services code point (DSCP) field. With the help of marking, traffic is identified for next action to achieve QoS. Each hop individually identified the incoming traffic on physical interface and provides the service level. We can mark traffic on Data link layer and on Network layer. Some methods of marking are as follows:

5.2.1 Data link layer marking

1. CoS value on IEEE 802.1p) [3]
   Three bits in IEEE 802.1 Q/P frame are reserved for QoS.
2. Multiprotocol Label Switching (MPLS) experimental (EXP) bits [3]
   Three bit field (MPLS EXP) is reserved for QoS purpose.
3. Frame Relay
   Forward explicit congestion notification (FECN), backward explicit congestion notification (BECN) and discard eligible (DE) fields are used for congestion management and congestion avoidance.

5.2.2 Network layer

1. IP precedence or DSCP on IP header
   IP precedence and DSCP uses 8-bit field ToS in IP header. IP precedence uses 3 most significant bits and DSCP uses 6 most significant bits. DSCP is backward compatible with IP precedence.
2. Source or destination IP address
   Source and destination IP address in IP can be used for marking the IP packets.

5.3 Per Hop Behaviour (PHB)

Per Hop Behavior (PHB) is a mechanism that is used by the DiffServ model to allocate the resource at each node in path. PHB is one that guarantees an x% allocation of network resource (Bandwidth, Delay, Reliability) to behavior aggregate at each node. This can be measured in variety of competing traffic conditions. This allocation of resource depends on business requirements. These PHBs are like building blocks and are grouped together to achieve QoS according to SLAs. PHBs are configured at each node in network in terms of buffer allocation and packet scheduling mechanisms. Figure 10 shows the bit pattern of PHB selector (DSCP field).

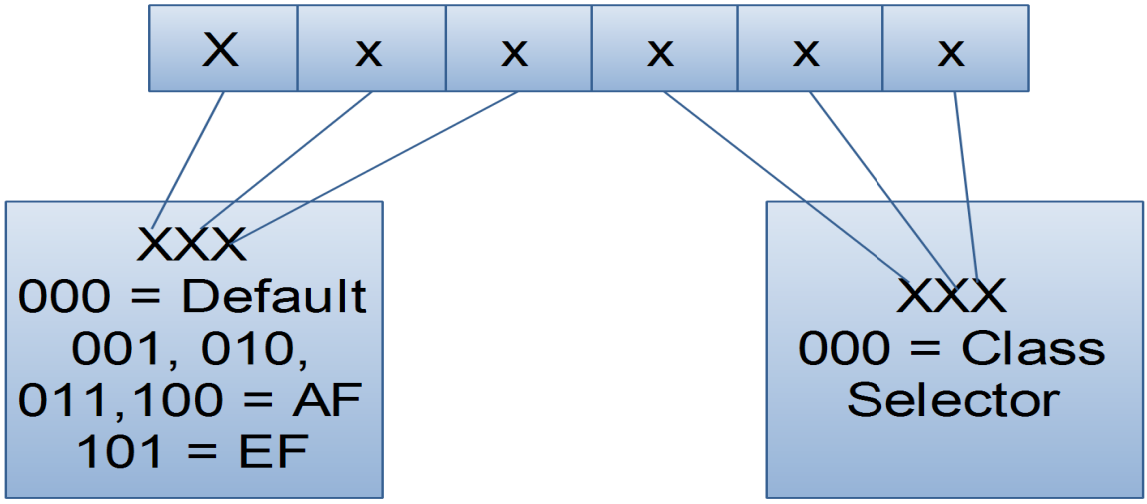


Figure 10: Per Hop Behaviour

IETF defines the following PHBs

1. Default PHB: Used for best-effort service.
2. Expedited Forwarding (EF): Used for low-delay service
3. Assured Forwarding (AF): Used for guaranteed bandwidth service
4. Class-selector PHB: Used for backward compatibility with non-DiffServ-compliant device

5.3.1 Expedited Forwarding PHB

Figure 11 defines the bit pattern of Expedited Forwarding used in DSCP. It provides the following functionalities.

1. Ensures the minimum end-to-end delay.
2. Provides guaranteed bandwidth.
3. Polices the bandwidth when congestion occurs.

Figure 11: EF bit pattern

5.3.2 Assured Forwarding PHB

Assured forwarding provide the following functionalities.

1. Provides some fix amount of bandwidth.
2. Allow extra bandwidth when available.

Figure 12 defines the bit pattern of Assured Forwarding used by DSCP in DiffServ model.

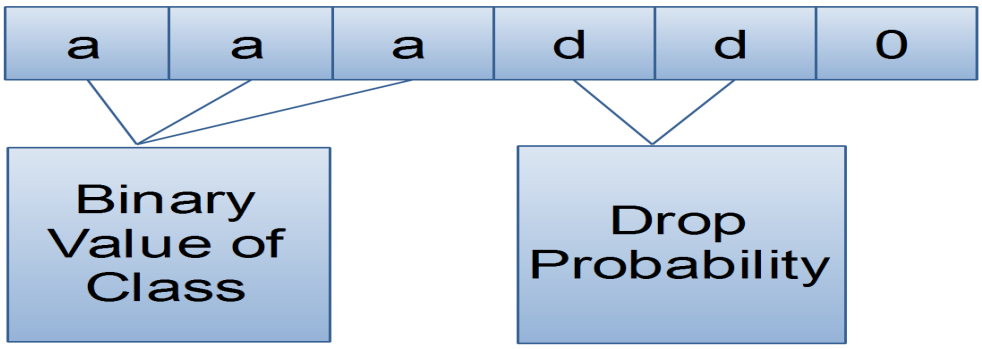


Figure 12: Assured Forwarding bit pattern

AF PHB is further divided into four classes with respect to the Drop Probability. These are AF1, AF2, AF3, and AF4. Each class has three level of drop probability which are LOW, MEDIUM and HIGH as shown if figure 13.

Class	Values		
AF1	001	dd	0
AF2	010	dd	0
AF3	011	dd	0
AF4	100	dd	0

Drop Prob. (dd)	Values	AF Value
LOW	01	AF11
Medium	10	AF12
High	11	AF13

Figure 13: AF PHB Values

AF Class	AF Code	Drop Probability	DSCP binary Value
Class1	AF11	LOW	001 01 0
	AF12	MEDIUM	001 10 0
	AF13	HIGH	001 11 0
Class 2	AF21	LOW	010 01 0
	AF22	MEDIUM	010 10 0
	AF23	HIGH	010 11 0
Class 3	AF31	LOW	011 01 0
	AF32	MEDIUM	011 10 0
	AF33	HIGH	011 11 0
Class 4	AF41	LOW	100 01 0
	AF42	MEDIUM	100 10 0
	AF43	HIGH	100 11 0

Table 1: AF drop strategy values

Table 1 describes in detail all the values of AF PHB, and its Drop probability respectively.

6 METHOD

The configuration of DiffServ QoS model over MPLS VPNs on real equipment has been achieved by using the network topology shown in figure 13. We will do step by step configurations and took the results and make the comparison of network behavior on different stages. We will use six routers in our scenario, four routers are in the provider domain and two are customer end routers as shown in figure 13. Two routers in the provider domain are P1 and P2, the functionality of these two are providing MPLS backbone. Two routers are the PE1 and PE2, PE1 is facing the Customer site1 and PE2 is facing Customer site2. PE1 and PE2 routers are providing connectivity to the customer routers and also providing the functionality of MPLS VPNs. Customer Edge routers will only run the normal routing protocols and they are isolated from the provider network.

Provider domain routers are running OSPF as an interior routing protocol. Customer's edge routers are running EIGRP as a routing protocol. The customer will not participate in the provider's routing because MPLS VPNs separate the customer routing from provider routing and it is done on provider's edge routers and customers feels it is like the dedicated circuit that is connecting to each site. All the route distribution is being done by the provider's edge routers. Virtual routers have been made in the form of VRF on the provider's edge routers that are only handling customer's routes separately. BGP peering is being configured between provider's edge routers to carry all the customer's routes from one site to another site.

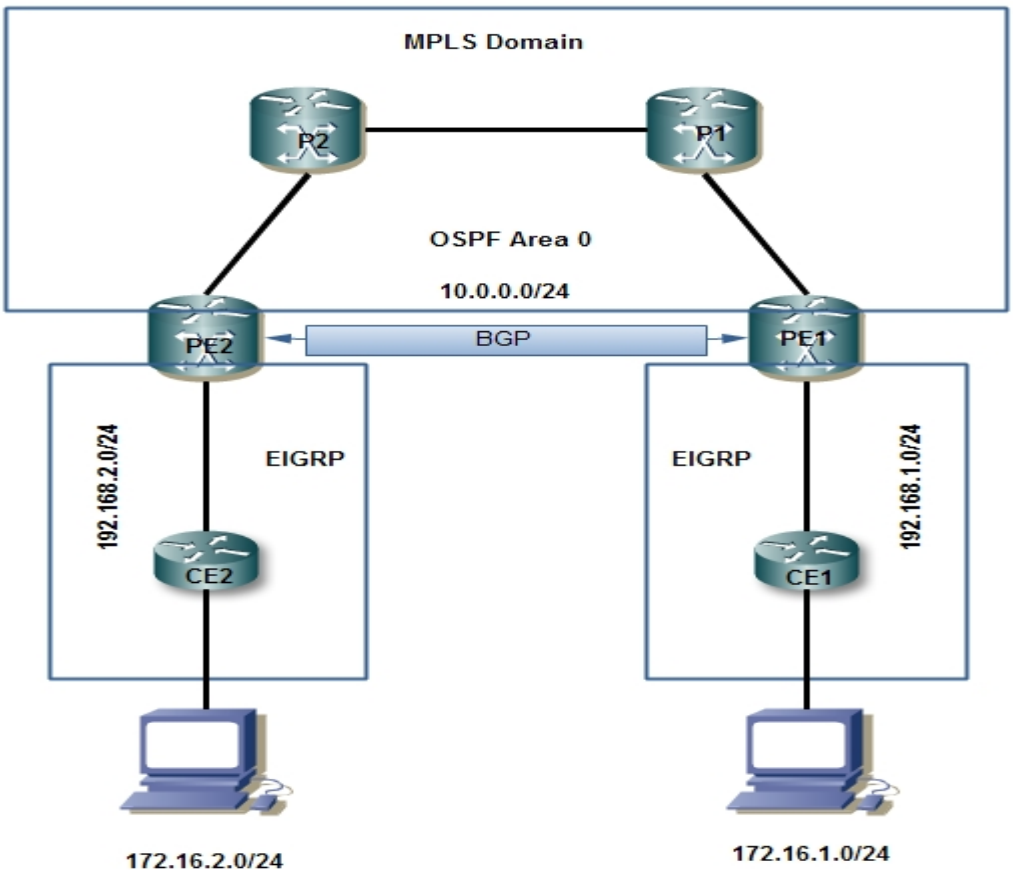


Figure 13: Network Topology

Figure 13 shows the network topology that is used for configuring DiffServ QoS model over MPLS VPNs environment.

At the last stage we make configuration to achieve the QoS. We use a DiffServ QoS model and categories the traffic into six classes. DiffServ QoS model is configured on customer's edge routers and also at the provider's edge router. Classification and Marking is done at the customer's routers. Provider's router are matching traffic on the basis of DSCP value and assigning bandwidth as described in table 2. All the routers configurations are shown in appendix A and all the outputs of routing tables and VRF tables are shown in appendix B.

Class Name	Match Criteria	DSCP Value	Assigned Bandwidth
Critical	BGP, OSPF, EIGRP, SNMP, Telnet, SSH	EF	5%
Video	RTSP, Vdolive	Af11	15%
Audio	RTP, RTCP, SIP, H323	AF21	15%
Mission Critical	Sqlserver, Sqlnet	AF31	10%
Web	http, Secure-http, SMTP, POP3, Secure-POP3, FTP, Secure-ftp, TFTP	AF41	5%
Scavenger	Any other	Default (000000)	50%

Table 2: Six Class QoS Model

6.1 Limitations and capacity

Voice and video use many encoding techniques to compress packets and decode at the receiving end. Different tools use different codec for voice and video traffic. In our case we use G.711 codec for voice and H.320 for the video call. G.711 use 64 Kbps for one voice call and H.320 use 384 Kbps for one video call [9]. Other codec consume different bandwidth and with the change of codec the results will be changed. G.711 and H.320 uses more bandwidth compared to other codec.

We assigned 15% bandwidth for voice calls and 15% bandwidth for video calls over 100mbps link. We are using G.711 and H.320 for audio and video calls. We can send 5 video calls and 30 audio calls in full congested network. When all the service class are congested and have no more capacity, our 6^{th} video and 31^{st} audio call will face bad quality. When we have no traffic on the network we can send 200 voice calls and 33 video calls on the network.

In a real world scenario, network topologies are more complex and more routers are involved in communication. In this thesis we use only six routers, if routers are increased in the path there will be some changes in results. Delay, jitter and packet loss can be increased. We practice how to use DiffServ QoS model over MPLS VPNs to achieve QoS and study the results of delay, jitter and packet loss using limited resources.

6.2 Traffic Generation

The traffic Generator is a special router that is made for QoS tests by Cisco Systems [3]. The traffic generator is used to generate different types of traffic patterns like TCP, UPD

and IP traffic. It can generate traffic streams and bursts of traffic. Traffic generator can generate all kind of IP traffic such as voice, video, real time streaming, web, file transfer, network management traffic. It can generate one or more traffic streams at a time. In this thesis we generate random IP traffic stream and increases the packet rate gradually. We put the traffic load on network by generating the random traffic while we make our tests on the base of User Datagram Protocol (UDP) traffic. In this thesis TGN is used for traffic generation and NQR is used for measuring the delay, jitter and packet loss. For testing the delay, jitter and packet loss we generate the traffic using NQR. TGN and NQR will be explained in section 6.3.

In this thesis we generate continuous IP traffic stream with two different methods. First we attach one traffic generator as shown in Figure 14. The traffic generator is attached with CE1 and CE2 routers via a switch and the switch has two VLANs to separate the traffic generator's interfaces. The continuous IP traffic stream is generated from the traffic generator's interface named Fast Ethernet 0/0. It passes through the CE1 router. It is received by the traffic generator on its interface named Fast Ethernet 0/1 as shown in figure 14.

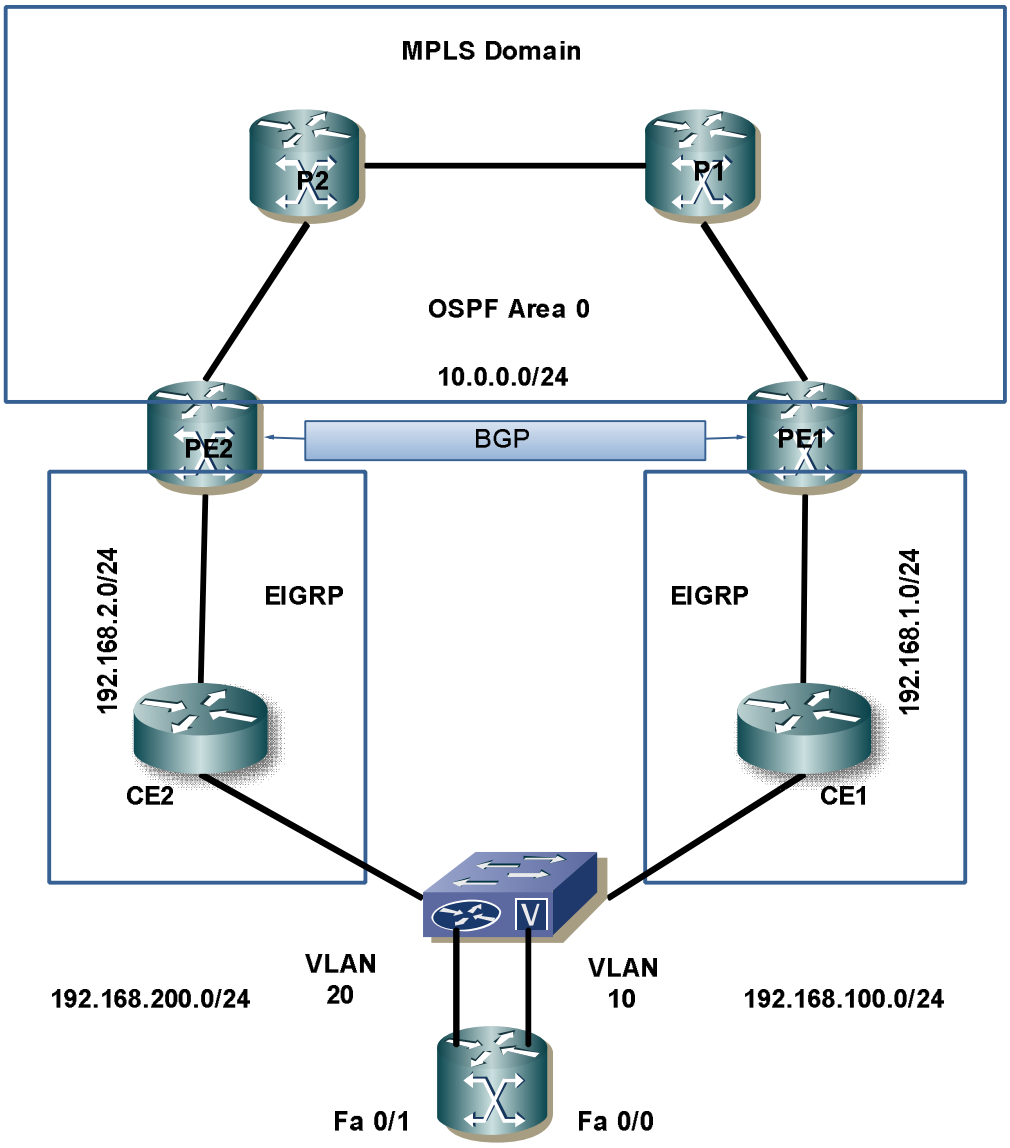


Figure 14: Topology with one Traffic Generator

In the next case we use three traffic generators and generate the same amount of traffic. In our second topology of traffic generation we replace CE1 and CE2 routers with traffic generators and reconfigure them as both customer edge routers and traffic generators, shown in Figure 15.

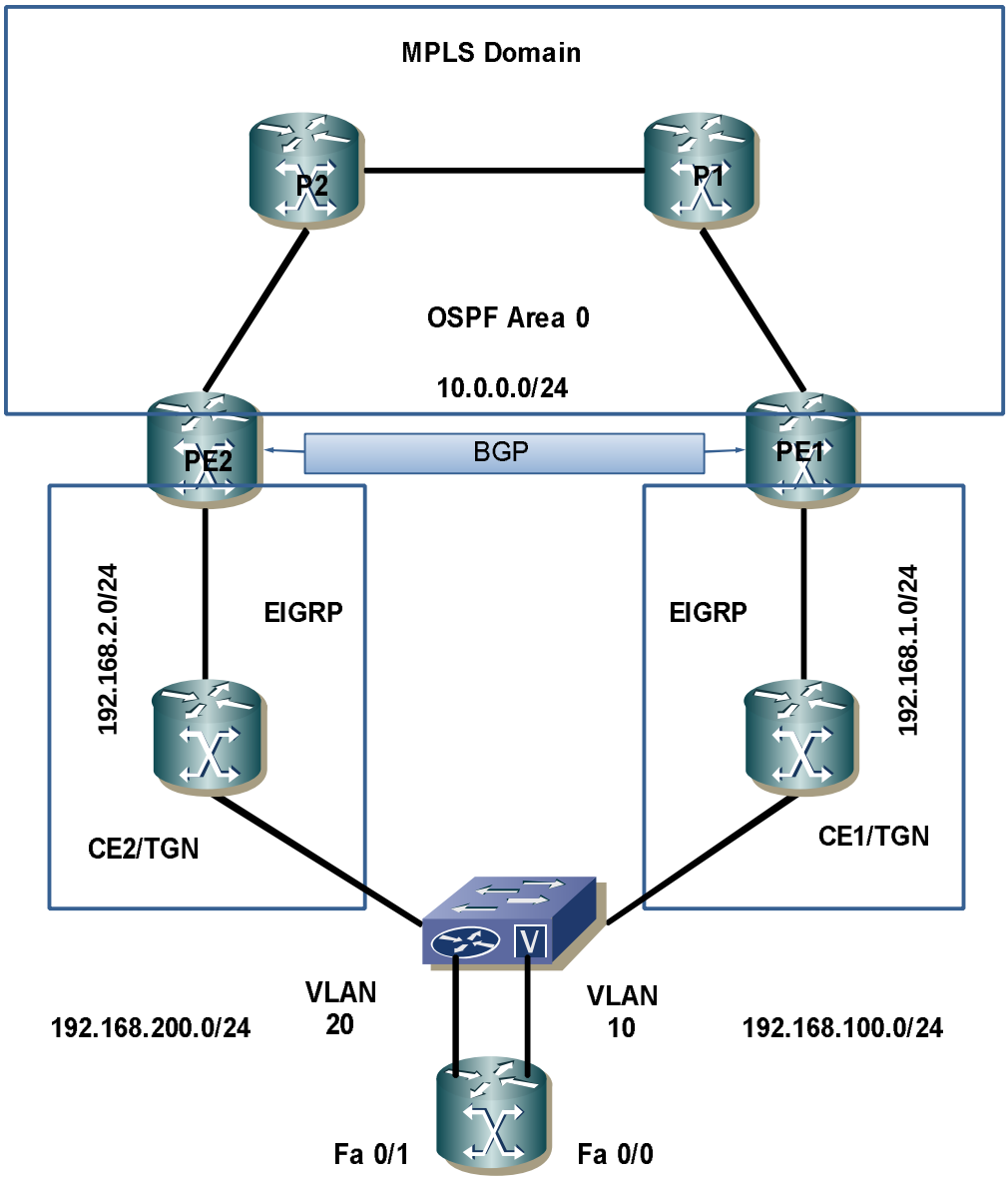


Figure 15: Topology with three traffic generator

6.3 Tools

6.3.1 TGN

TGN is an internetworking operating system (IOS) based program that runs on a pagent router (Cisco traffic generator). It is used to generate different kind of traffics patterns. We use it to generate continuous IP traffic stream. It is required almost all header fields information to generate traffic. We have to put layer 2, layer 3 for IP traffic generation. We can generate different types of traffic by providing the different header information. We can change TCP/UDP source and destination ports for different traffic generation. When we generate traffic we have to give the MAC address of the next hope router to avoid broadcasts. The traffic will send to a given MAC destination address (the next hope address) and the traffic stream traverses the network and ends up on traffic generation router. In this thesis we generate an IP traffic stream for background network traffic. It is continuous in flow and

periodic in packet generation. Packet size is maintained 100 bytes in length and traffic load is maintained by packets per seconds (PPs) in IP stream. We increase PPs gradually start from 3000 PPs ends up 30,000 PPs. We take the result statistics at each increment of 3000 PPs. This increment in PPs is not automatic each time we have to stop IP traffic stream, reconfigure packet rate and start it to take the results. Following are the commands used on Cisco pagent router for generating this background traffic.

```
TGN#tgn

TGN(TGN:OFF,Vo0:none)#fastethernet 0/0

TGN(TGN:OFF,Vo0:none)#add ip

TGN(TGN:OFF,Fa0/0:1/1)#name IP Stream

TGN(TGN:OFF,Fa0/0:1/1)#L2-dest-addr xxxx.xxxx.xxxx

TGN(TGN:OFF,Fa0/0:1/1)#L2-src-addr xxxx.xxxx.xxxx

TGN(TGN:OFF,Fa0/0:1/1)#L3-src-addr x.x.x.x

TGN(TGN:OFF,Fa0/0:1/1)#L3-dest-addr x.x.x.x

TGN(TGN:OFF,Fa0/0:1/1)#rate 3000

TGN(TGN:OFF,Fa0/0:1/1)#length 100

TGN(TGN:OFF,Fa0/0:1/1)#on

TGN(TGN:ON,Fa0/0:1/1)#end

TGN#
```

6.3.2 NQR

A network quality reporter (NQR) is a simple IOS based pagent router tool. NQR is used to measure the quality of the network by generating its own traffic. It is used to measure end-to-end delay, jitter and packet loss. One interface of the router is used to send the traffic and one interface is used to capture the traffic. We configured the traffic generator's one interface to generate traffic and the other to capture it. NQR supports IP based protocols like UDP, TCP, ICMP, IP, and IGMP. We generate UDP traffic stream for taking results because audio and video traffic use UDP at transport layer. Values of delay, jitter, and packet loss are taken at each increment of background traffic of 3000 PPs. In NQR we maintain the datagram size of 100 bytes and packet rate is maintained of 500 PPs at each load of back ground traffic. Followings are the configuration commands for measuring delay, jitter and packet loss.

```
TGN#nqr

TGN(NQR:OFF,Vo0:none)#add udp

TGN(NQR:OFF,Vo0:1/1)#fastethernet 0/0
```

```
TGN(NQR:OFF,Fa0/0:1/1)#name NQR
TGN(NQR:OFF,Fa0/0:1/1)#fastethernet 0/1 capture on
TGN(NQR:OFF,Fa0/0:1/1)#L2-dest-addr xxxx.xxxx.xxxx
TGN(NQR:OFF,Fa0/0:1/1)#L2-src-addr xxxx.xxxx.xxxx
TGN(NQR:OFF,Fa0/0:1/1)#L3-src-addr x.x.x.x
TGN(NQR:OFF,Fa0/0:1/1)#L3-dest-addr x.x.x.x
TGN(NQR:OFF,Fa0/0:1/1)#rate 500
TGN(NQR:OFF,Fa0/0:1/1)#length 100
TGN(NQR:OFF,Fa0/0:1/1)#start
TGN(NQR:ON,Fa0/0:1/1)#show stats
TGN(NQR:ON,Fa0/0:1/1)#end
TGN#
```

7 Results of Test, Measurement or Simulation

7.1 Simple IP Network

In our first case we have a simple IP Network in which we only used the EIGRP routing protocol for end-to-end connectivity and the value of delay, jitter and packet loss with respect to packets per second are shown in table 3.

PPS	Delay	Jitter	Packet Loss
3000	0.000289	0.000037	0
6000	0.000531	0.000062	0.09616
9000	0.000835	0.000533	1.9767
12000	0.001377	0.000804	2.3996
15000	0.001641	0.001203	2.9750
18000	0.002025	0.001481	6.6693
21000	0.002135	0.001729	19.4549
24000	0.002439	0.001934	40.7475
27000	0.002557	0.002065	40.7299
30000	0.002721	0.002161	40.0657

Table 3: IP Network Environment Values

7.2 MPLS Enabled Network

In our second case we configured a MPLS enabled network in the service provider domain (P1, P2, PE1 and PE2.) and took the results of delay, jitter, and packet loss as shown in table 4.

PPS	Delay	Jitter	Packet loss
3000	0.000290	0.000042	0
6000	0.000339	0.000084	0
9000	0.000735	0.000197	4.2952
12000	0.000709	0.000287	6.4653
15000	0.000976	0.000255	5.7508
18000	0.001104	0.000485	8.3088
21000	0.001599	0.000774	13.6772
24000	0.001868	0.000966	14.8716
27000	0.001976	0.001471	17.2672
30000	0.002079	0.001598	20.2599

Table 4: MPLS Enabled Network Environment Values

7.3 MPLS Enabled VPN Network

In our third case we configured MPLS VPNs on PE1 and PE2 routers and take the results of delay, jitter, and packet loss as shown in table 5.

PPS	Delay	Jitter	Packet loss
3000	0.000337	0.000060	0
6000	0.000340	0.000063	0
9000	0.000528	0.000189	3.9402
12000	0.000675	0.000306	6.5109
15000	0.000874	0.000486	11.67
18000	0.000982	0.000657	12.8327
21000	0.001421	0.001043	15.65
24000	0.001599	0.001189	18.8876
27000	0.001816	0.001685	21.0337
30000	0.002248	0.001753	20.9839

Table 5: MPLS VPN Enabled Network Environment Values

7.4 MPLS Enabled VPN with QoS

In the fourth case we configured the MPLS VPN with a DiffServ QoS model and took the results in two different ways. First we obtained the result by generating the traffic from three routers in which CE1, CE2 and one external traffic generator routers were included. Secondly we generated the traffic only from the external traffic generator router and took the results.

7.4.1 Results from three Traffic generators

Table 6 shows the results in MPLS VPN with a DiffServ QoS model and this traffic is generated by CE1, CE2 and one external router as shown in figure 15.

PPS	Delay	Jitter	Packet Loss
3000	0.000880	0.000119	0.0000
6000	0.000887	0.000115	0.0000
9000	0.000883	0.000120	0.0000
12000	0.000888	0.000123	0.0000
15000	0.000890	0.000124	0.0000
18000	0.000883	0.000118	0.0000
21000	0.000893	0.000124	0.0000
24000	0.000880	0.000120	0.0000
27000	0.000896	0.000129	0.0000
30000	0.000896	0.000126	0.0000
150,000	0.000906	0.000121	0.0000
300,000	0.000903	0.000125	0.0000

Table 6: MPLS VPN with DiffServ Enabled Network Environment Values*3

7.4.2 Results from one Traffic generator

Table7 shows the Results of MPLS VPNs with a DiffServ QoS model enabled network environment. This traffic is generated from one external router as shown in figure 14.

PPS	Delay	Jitter	Packet Loss
3000	0.000653	0.000056	0.0000
6000	0.000650	0.000065	0.0000
9000	0.000660	0.000061	0.0000
12000	0.000662	0.000061	0.0000
15000	0.000649	0.000057	0.0000
18000	0.000656	0.000058	0.0000
21000	0.000651	0.000058	0.0000
24000	0.000651	0.000059	0.0000
27000	0.000657	0.000058	0.0000
30000	0.000655	0.000059	0.0000
150,000	0.000655	0.000059	0.0000
300,000	0.000655	0.000059	0.0000

Table 7: MPLS VPN with DiffServ Enabled Network Environment Values*1

7.5 Result Analysis

In order to analyze results, we gathered the values of delay, jitter, and packet loss from our results at one place and made the comparison in different cases.

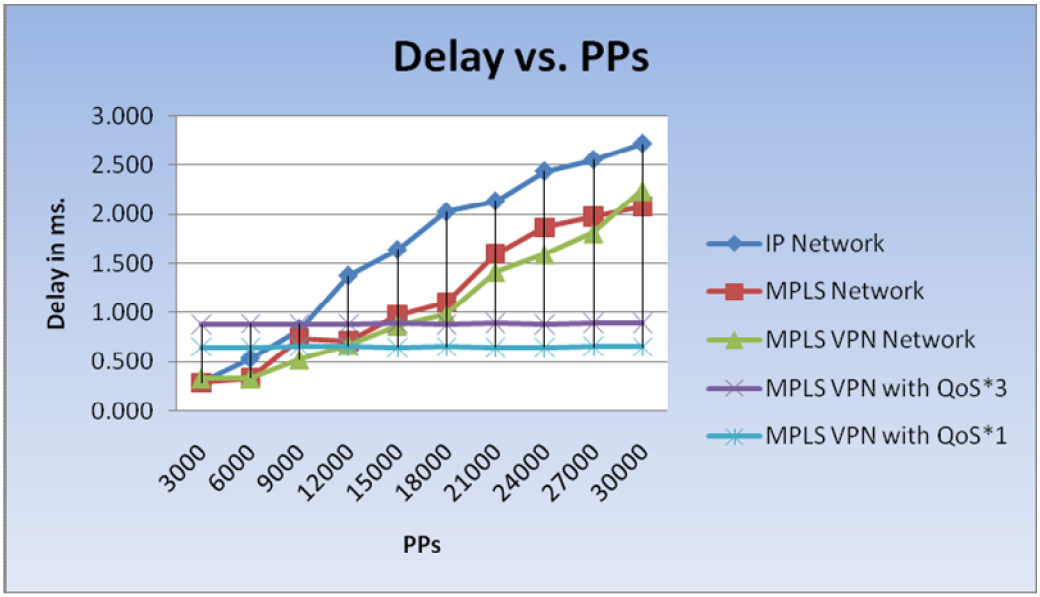
7.5.1 Delay

Table 8 shows the details of delay values in different network cases. This table shows PPs against the delay in milliseconds. Traffic load starts from 3000 PPs and continues in increment of 3000 PPs ending up at 30,000 PPs. Delay in simple IP network varies between 0.289ms to 2.721ms whereas delay in MPLS network varies between 0.290ms to 2.079ms. Delay in MPLS VPN network varies between 0.337ms to 2.248ms. Delay in MPLS network with QoS*3 (traffic generation from three routers as shown in figure 15) varies between 0.880ms to 0.896ms and delay in MPLS VPN with QoS*1 (traffic generation from one router as shown in figure 14) varies between 0.649ms to 0.657ms.

PPS	Delay in ms.				
	IP	MPLS	MPLS VPN	MPLS VPN with QoS*3	MPLS VPN with QoS*1
3000	0.289	0.290	0.337	0.880	0.653
6000	0.531	0.339	0.340	0.887	0.650
9000	0.835	0.735	0.528	0.883	0.660
12000	1.377	0.709	0.675	0.888	0.662
15000	1.641	0.976	0.874	0.890	0.649
18000	2.025	1.104	0.982	0.883	0.656
21000	2.135	1.599	1.421	0.893	0.651
24000	2.439	1.868	1.599	0.880	0.651
27000	2.557	1.976	1.816	0.896	0.657
30000	2.721	2.079	2.248	0.896	0.655

Table 8: Delay Comparison

Graph 1 shows delay versus PPs, as we see in the simple IP network case the delay is increasing with the traffic increase on the network. We can see in graph 1 that the simple IP network case has more delay compared to the MPLS network or the MPLS VPN enabled network. When we configure DiffServ QoS model over MPLS VPN the delay value is almost constant for different traffic load (PPs).In MPLS VPN with QoS*3 case the average packet delay is 0.9ms. In MPLS VPN with QoS*1 case the delay is almost 0.65ms.



Graph 1: Delay vs. PPs

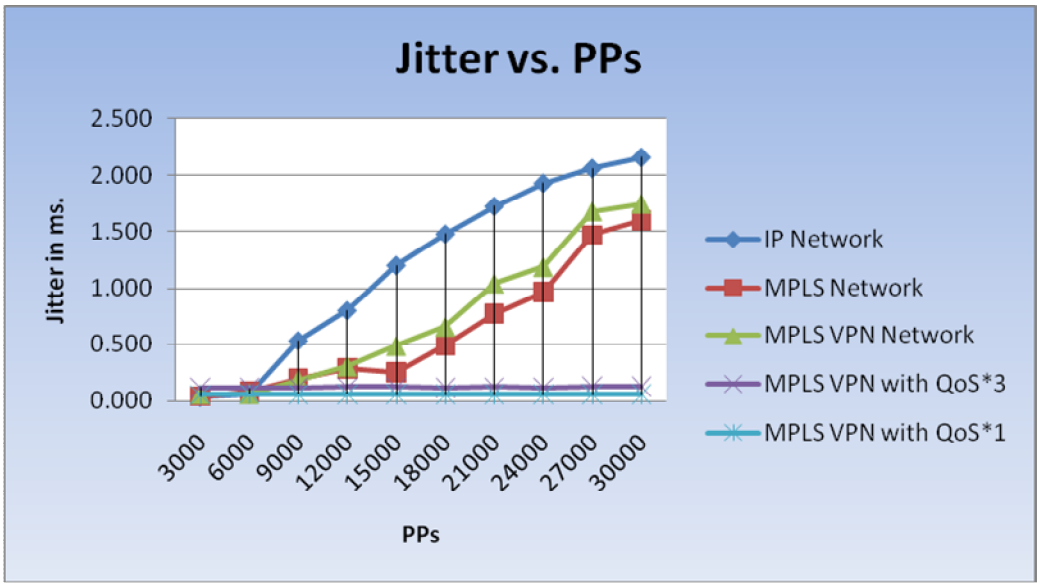
7.5.2 Jitter:

In table 9 values of jitter in different cases are gathered to make a more effective analysis. In table 9 all the values are shown with respect to PPs. PPs start from 3000 and goes up till 30,000 with the increment of 3000PPS each time. In the simple IP network jitter varies between 0.037ms to 2.161ms. In the MPLS network jitter varies between 0.042ms to 1.598ms. In the MPLS VPN case jitter varies between0.060ms to 1.753ms. In the MPLS VPN with QoS*3 jitter varies between 0.0119ms to 0.126ms. In the MPLS VPN with QoS*1 jitter varies between 0.056ms to 0.059ms.

Graph 2 shows jitter versus PPs, as we can see in the case of simple IP network jitter increases with traffic load. We can see in graph 2 that the simple IP network has more jitter compared to the MPLS network or MPLS VPN enabled network. When we configured the DiffServ QoS model over MPLS VPN the jitter value is almost constant for increasing PPs. MPLS VPN with QoS*3 jitter is almost 0.120ms. MPLS VPN with QoS*1 jitter is almost 0.60ms. The variation with QoS cases is very low.

PPS	Jitter in ms.				
	IP	MPLS	MPLS VPN	MPLS VPN with QoS*3	MPLS VPN with QoS*1
3000	0.037	0.042	0.060	0.119	0.056
6000	0.062	0.084	0.063	0.115	0.065
9000	0.533	0.197	0.189	0.120	0.061
12000	0.804	0.287	0.306	0.123	0.061
15000	1.203	0.255	0.486	0.124	0.057
18000	1.481	0.485	0.657	0.118	0.058
21000	1.729	0.774	1.043	0.124	0.058
24000	1.934	0.966	1.189	0.120	0.059
27000	2.065	1.471	1.685	0.129	0.058
30000	2.161	1.598	1.753	0.126	0.059

Table 9: Jitter comparison



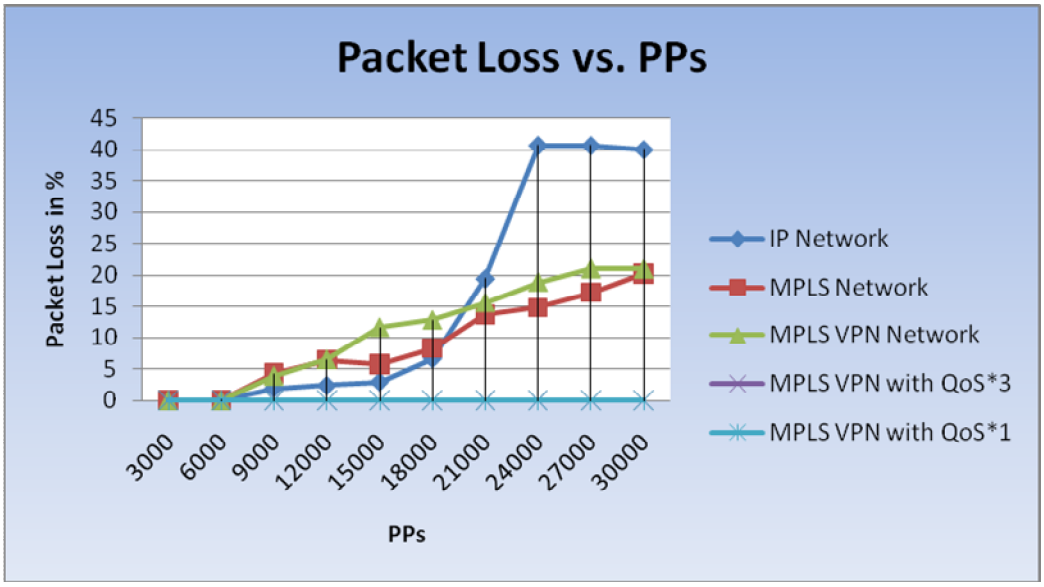
Graph 2: Jitter vs. PPs

7.5.3 Packet Loss

Table 10 show details of packet loss in different network cases. Table 9 represents the values of packet loss by percentage against PPs. Traffic load starts from 3000 PPs and continues in increments of 3000 PPs ending up on 30,000 PPs. In the simple IP network case packet loss varies from 0% to 40.0657%. In the MPLS network case packet loss varies 0% to 20.2599%. In the MPLS VPN network case packet loss changes from 0% to 20.9839%. In the MPLS VPN with QoS*3 and MPLS VPN with QoS*1 cases packet loss is 0%.

PPS	Packet loss %				
	IP	MPLS	MPLS VPN	MPLS VPN with QoS*3	MPLS VPN with QoS*1
3000	0	0	0	0.0000	0.0000
6000	0.09616	0	0	0.0000	0.0000
9000	1.9767	4.2952	3.9402	0.0000	0.0000
12000	2.3996	6.4653	6.5109	0.0000	0.0000
15000	2.9750	5.7508	11.67	0.0000	0.0000
18000	6.6693	8.3088	12.8327	0.0000	0.0000
21000	19.4549	13.6772	15.65	0.0000	0.0000
24000	40.7475	14.8716	18.8876	0.0000	0.0000
27000	40.7299	17.2672	21.0337	0.0000	0.0000
30000	40.0657	20.2599	20.9839	0.0000	0.0000

Table 10: Packet loss comparison



Graph 3: Packet Loss vs. PPs

Graph 3 shows the relationship between packet losses in percentage with respect to PPs. In the case of a simple IP network packet loss is higher than for the MPLS enabled network and MPLS VPN enabled network. Graph 3 clearly shows that after configuring DiffServ QoS model packet loss is 0% independent if the traffic is generated from three routers or from one router.

Conclusion

MPLS VPN is a new choice for WAN connectivity [3]. It combines the features of Private WAN connectivity (Frame Relay, ATM, and Leased Lines) and layer2 VPNs. MPLS VPNs reduces the complexity of network operations. It also reduces the cost to manage the network operations. By using DiffServ QoS model it is easy to manage network resources and getting the maximum utilization from available resources. Enterprise networks send all video, audio traffic as well as elastic data traffic over the same network infrastructure. We can use DiffServ QoS model to gain the quality of experience for end user in MPLS VPN environment. It is necessary for the customer's network and service provider's network to better manage the resources. DiffServ QoS model itself does not create bandwidth but it manages the available bandwidth. It is used for well-defined capacity planning and overall application governance process. Network engineers can make traffic management decisions by analyzing network capacity and application's requirement of resources.

MPLS is a fast packet switching technology and reduces the end-to-end delay. The DiffServ QoS model is more effective and scalable than the IntServ QoS model. We can get better results in MPLS VPN network environment by using a DiffServ QoS model. Without DiffServ QoS model in MPLS VPN network environment delay, jitter, and packet loss are rising with the increase of traffic on the network. With the configuration of DiffServ QoS model, it provides almost constant delay, jitter, and packet loss in all different traffic loads bounded by the limitations. Limitations are considered by the means of allocated resources for specific traffic class. Scalable video and audio service with good quality, over the enterprise network using MPLS VPNs together with DiffServ QoS model, can be provided.

Results are taken from the tests using NQR. In MPLS VPN environment delay was calculated between "0.290ms to 2.079ms", jitter was"0.060ms to 1.753ms", and packet loss was"0% to 20.9839%". After using the DiffServ QoS model delay was calculated between "0.649ms to 0.662ms", jitter was between "0.056ms to 0.061ms", and packet loss was 0%. In MPLS VPN with DiffServ QoS model environment, the values of delay, jitter and packet loss in different traffic loads have very low variation and are almost constant.

8 References

[1]. J. Lawrence, *Designing multiprotocol label switching networks, Communications Magazine,* IEEE, Volume 39, Issue 7, July 2001.

[2]. *Luc De Ghein, MPLS Fundamentals, Cisco Systems,* Cisco Press 800 East 96th Street Indianapolis, ISBN: 1-58705-197-4, 2007.

[3]. Vivek Alwayn, *Advanced MPLS design and Implementation, Cisco Systems,* Cisco press 201 west 103rd Street Indianapolis, 2001.

[4]. Ivan Pepelnjak , Jim Guichard, *MPLS and VPN Architecture, Cisco Systems,* Cisco press 201 West 103rd Street Indianapolis, March 2001

[5]. How to configure MPLS VPN over ATM using cell mode MPLS with BGP or RIPv2 on the customer site: "http://www.networkworld.com/community/node/30050 " (last access date November 2010)

[6]. Amir Ranjbar, "CCNP ONT Official Exam Certification Guide", First Edition, Published by: Cisco Press 800 East 96th Street Indianapolis, IN 46240 USA 2007, ISBN-10: 1-58720-176-3, ISBN-13: 978-1-58720-176-9.

[7]. DiffServ: Scalable End-to-End QoS Model on Cisco Systems site: "http://www.cisco.com/en/US/technologies/tk543/tk766/technologies_white_paper09186a00800a3e2f.html" (last access November 2010).

[8]. R. Braden, D. Clark, S. Shenker, *Integrated Services in the Internet Architecture: an Overview,* RFC 1633, June 1994.

[9]. Voice Over IP − Per Call Bandwidth Consumption, Cisco Systems site, Document ID: 7934, on site: "http://www.cisco.com/application/pdf/paws/7934/bwidth_consume.pdf" (last access November 2010).

[10]. H.320 Gateway to H.323 Gatekeeper Video Call Flow, Cisco Systems site, Document ID:72056.
"http://www.cisco.com/en/US/tech/tk1077/technologies_configuration_example09186a00807ca099.shtml#bandwidth" (last access November 2010).

[11]. Mohamed EL Hachimi,Marc-Andr´e Breton,Maria Bennani, *Efficient QoS implementation for MPLS VPN, 22nd International Conference on Advanced Information Networking and Applications – Workshops, IEEE, Issue Date: 25-28 March 2008.*

[12]. Fan Ya-qin, Wang Lin-zhu, Zhang Li-cui. *Computational Intelligence and Design, 2008. ISCID '08. International Symposium, IEEE, Issued* at Wuhan, Issued on Oct. 2008.

[13]. Fan Ya-qin, Wang Lin-zhu, Zhang Li-cui,*Research for QoS of MPLS VPN based on Log-infinitely Divisible Cascades. IEEE, 2008 International Symposium on Computational Intelligence and Design,* Issued Date: Oct. 2008.

[14]. C. Huang, and Vishal Sharma, *"Building Reliable MPLS Networks Using a Path Protection Mechanism,"* IEEE Communication Magazine, Issued on Mar. 2002, pp. 156 – 162.

[15]. Antonis Nikitakis, Antonis Nikitakis. *"A Multi Gigabit FPGA-based 5-tuple classification system"*. IEEE Communications Society at ICC, issued on 2008.

[16]. Brian Morgan, Neil Lovering "CCNP ISCW Official Exam Certification Guide", First Edition Published by: Cisco Press 800 East 96th Street Indianapolis, IN 46240 USA 2007, ISBN-13: 978-1-58720-150-9, ISBN-10: 1-58720-150-x.

9 Appendices

In appendix A and appendix B have some standard configuration of Cisco routers, and may have plagiarism.

9.1 Appendix A
Routers Configurations

P1#sh run

Building configuration...

Current configuration : 1718 bytes

!

version 12.4

service timestamps debug datetime msec

service timestamps log datetime msec

no service password-encryption

!

hostname P1

!

boot-start-marker

boot-end-marker

!

!

no aaa new-model

!

resource policy

!

memory-size iomem 5

mmi polling-interval 60

```
no mmi auto-configure
no mmi pvc
mmi snmp-timeout 180
ip subnet-zero
ip cef
!
!
!
voice-card 0
!
!
!
class-map match-any critical
 match ip dscp ef
class-map match-any mission_critical
 match ip dscp af31
class-map match-any web
 match ip dscp af41
class-map match-any video
 match ip dscp af11
class-map match-any voice
 match ip dscp af21
class-map match-any scavenger
 match ip dscp default
!
!
policy-map weblinx
```

```
 class critical
 bandwidth percent 5
 class video
 bandwidth percent 15
 class voice
 bandwidth percent 15
 class mission_critical
 bandwidth percent 10
 class web
 bandwidth percent 5
 class scavenger
 police rate percent 20
   exceed-action drop
!
!
interface Loopback0
 ip address 10.0.101.1 255.255.255.255
!
interface FastEthernet0/0
 ip address 10.0.1.1 255.255.255.0
 duplex auto
 speed auto
 mpls ip
 service-policy output weblinx
!
interface FastEthernet0/1
 ip address 10.0.2.1 255.255.255.0
```

```
 duplex auto
 speed auto
 mpls ip
 service-policy output weblinx
!
interface Serial0/1/0
 no ip address
 shutdown
 no fair-queue
 clock rate 125000
!
interface Serial0/1/1
 no ip address
 shutdown
 clock rate 125000
!
router ospf 1
 log-adjacency-changes
 network 10.0.0.0 0.255.255.255 area 0
!
ip classless
!
!
ip http server
no ip http secure-server
!
!
```

```
!
!
mpls ldp router-id Loopback0 force
!
control-plane
!
!
line con 0
line aux 0
line vty 0 4
 login
!
end
```

P2#sh run

Building configuration...

```
Current configuration : 1718 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname P2
!
```

```
boot-start-marker
boot-end-marker
!
!
no aaa new-model
!
resource policy
!
memory-size iomem 5
mmi polling-interval 60
no mmi auto-configure
no mmi pvc
mmi snmp-timeout 180
ip subnet-zero
ip cef
!
!
voice-card 0
!
!
class-map match-all critical
 match ip dscp ef
class-map match-all mission_critical
 match ip dscp af31
class-map match-all web
 match ip dscp af41
class-map match-all video
```

```
 match ip dscp af11
class-map match-all voice
 match ip dscp af21
class-map match-all scavenger
 match ip dscp default
!
!
policy-map weblinx
  class critical
  bandwidth percent 5
  class video
  bandwidth percent 15
  class voice
  bandwidth percent 15
  class mission_critical
  bandwidth percent 10
  class web
  bandwidth percent 5
  class scavenger
  police rate percent 20
    exceed-action drop
!

interface Loopback0
 ip address 10.0.102.1 255.255.255.255
!
interface FastEthernet0/0
```

```
 ip address 10.0.1.2 255.255.255.0
 duplex auto
 speed auto
 mpls ip
 service-policy output weblinx
!
interface FastEthernet0/1
 ip address 10.0.3.2 255.255.255.0
 duplex auto
 speed auto
 mpls ip
 service-policy output weblinx
!
interface Serial0/1/0
 no ip address
 shutdown
 no fair-queue
 clock rate 125000
!
interface Serial0/1/1
 no ip address
 shutdown
 clock rate 125000
!
router ospf 1
 log-adjacency-changes
 network 10.0.0.0 0.255.255.255 area 0
```

```
!
ip classless
!
!
ip http server
no ip http secure-server
!
!
mpls ldp router-id Loopback0 force
!
control-plane
!
!
!
line con 0
line aux 0
line vty 0 4
 login
!
end
```

PE1#sh run

Building configuration...

```
Current configuration : 2397 bytes
!
version 12.4
service timestamps debug datetime msec
```

```
service timestamps log datetime msec
no service password-encryption
!
hostname PE1
!
boot-start-marker
boot-end-marker
!
!
no aaa new-model
!
resource policy
!
memory-size iomem 5
mmi polling-interval 60
no mmi auto-configure
no mmi pvc
mmi snmp-timeout 180
ip subnet-zero
ip cef
!
!
ip vrf weblinx
 rd 100:1
 route-target export 1:100
 route-target import 1:100
!
```

```
!
voice-card 0
!
!
!
!
class-map match-any critical
 match ip dscp ef
class-map match-any mission_critical
 match ip dscp af31
class-map match-any web
 match ip dscp af41
class-map match-any video
 match ip dscp af11
class-map match-any voice
 match ip dscp af21
class-map match-any scavenger
 match ip dscp default
!
!
policy-map weblinx
  class critical
  bandwidth percent 5
  class video
  bandwidth percent 15
  class voice
  bandwidth percent 15
```

```
  class mission_critical
  bandwidth percent 10
  class web
  bandwidth percent 5
  class scavenger
  police rate percent 20
    exceed-action drop
!
!
interface Loopback0
 ip address 10.0.103.1 255.255.255.255
!
interface FastEthernet0/0
 ip vrf forwarding weblinx
 ip address 192.168.1.3 255.255.255.0
 duplex auto
 speed auto
 service-policy output weblinx
!
interface FastEthernet0/1
 ip address 10.0.2.3 255.255.255.0
 duplex auto
 speed auto
 mpls ip
 service-policy output weblinx
!
interface Serial0/1/0
```

```
 no ip address
 shutdown
 clock rate 125000
!
interface Serial0/1/1
 no ip address
 shutdown
 clock rate 125000
!
router eigrp 100
 auto-summary
 !
 address-family ipv4 vrf weblinx
 redistribute bgp 100 metric 100000 0 255 1 1508
 network 192.168.1.0
 no auto-summary
 autonomous-system 1
 exit-address-family
!
router ospf 1
 log-adjacency-changes
 network 10.0.0.0 0.255.255.255 area 0
!
router bgp 100
 no synchronization
 bgp log-neighbor-changes
 neighbor 10.0.104.1 remote-as 100
```

```
 neighbor 10.0.104.1 update-source Loopback0
 no auto-summary
 !
 address-family vpnv4
 neighbor 10.0.104.1 activate
 neighbor 10.0.104.1 send-community both
 exit-address-family
 !
 address-family ipv4 vrf weblinx
 redistribute eigrp 1
 no auto-summary
 no synchronization
 exit-address-family
!
ip classless
!
!
ip http server
no ip http secure-server
!
!
mpls ldp router-id Loopback0 force
!
control-plane
!
!
line con 0
```

line aux 0

line vty 0 4

login

!

end

PE2#sh run

Building configuration...

Current configuration : 2397 bytes

!

version 12.4

service timestamps debug datetime msec

service timestamps log datetime msec

no service password-encryption

!

hostname PE2

!

boot-start-marker

boot-end-marker

!

!

no aaa new-model

!

resource policy

!

mmi polling-interval 60

no mmi auto-configure

```
no mmi pvc
mmi snmp-timeout 180
ip subnet-zero
ip cef
!
!
ip vrf weblinx
 rd 100:1
 route-target export 1:100
 route-target import 1:100
!
!
voice-card 0
!
!
!
class-map match-any critical
 match ip dscp ef
class-map match-any mission_critical
 match  dscp 31
 match ip dscp af31
class-map match-any web
 match ip dscp af41
class-map match-any video
 match ip dscp af11
class-map match-any voice
 match ip dscp af21
```

```
class-map match-any scavenger
 match ip dscp default
!
!
policy-map weblinx
  class critical
  bandwidth percent 5
  class video
  bandwidth percent 15
  class voice
  bandwidth percent 15
  class mission_critical
  bandwidth percent 10
  class web
  bandwidth percent 5
  class scavenger
  police rate percent 20
    exceed-action drop
!
!
!
interface Loopback0
 ip address 10.0.104.1 255.255.255.255
!
interface FastEthernet0/0
 ip vrf forwarding weblinx
 ip address 192.168.2.4 255.255.255.0
```

```
 duplex auto
 speed auto
 service-policy output weblinx
!
interface FastEthernet0/1
 ip address 10.0.3.4 255.255.255.0
 duplex auto
 speed auto
 mpls ip
 service-policy output weblinx
!
interface Serial0/1/0
 no ip address
 shutdown
 clock rate 125000
!
interface Serial0/1/1
 no ip address
 shutdown
 clock rate 125000
!
router eigrp 100
 no auto-summary
 !
 address-family ipv4 vrf weblinx
 redistribute bgp 100 metric 100000 0 255 1 1508
 network 192.168.2.0
```

```
 no auto-summary
 autonomous-system 1
 exit-address-family
!
router ospf 1
 log-adjacency-changes
 network 10.0.0.0 0.255.255.255 area 0
!
router bgp 100
 no synchronization
 bgp log-neighbor-changes
 neighbor 10.0.103.1 remote-as 100
 neighbor 10.0.103.1 update-source Loopback0
 no auto-summary
 !
 address-family vpnv4
 neighbor 10.0.103.1 activate
 neighbor 10.0.103.1 send-community both
 exit-address-family
 !
 address-family ipv4 vrf weblinx
 redistribute eigrp 1
 no auto-summary
 no synchronization
 exit-address-family
!
ip classless
```

```
!
!
ip http server
no ip http secure-server
!
!
mpls ldp router-id Loopback0 force
!
control-plane
!
!
!
line con 0
line aux 0
line vty 0 4
 login
!
end
```

```
CE1#sh run
Building configuration...

Current configuration : 2069 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
```

```
!
hostname CE1
!
boot-start-marker
boot-end-marker
!
!
no aaa new-model
memory-size iomem 5
!
!
ip cef
!
!
ip host PAGENT-SECURITY-V3 21.63.1.51 90.34.0.0
!
multilink bundle-name authenticated
!
!
voice-card 0
 no dspfarm
!
!
class-map match-any critical
 match protocol bgp
 match protocol ospf
 match protocol eigrp
```

```
 match protocol snmp
 match protocol telnet
 match protocol ssh
class-map match-any mission_critical
 match protocol sqlserver
 match protocol sqlnet
class-map match-any web
 match protocol http
 match protocol secure-http
 match protocol secure-ftp
 match protocol secure-pop3
 match protocol ftp
 match protocol tftp
 match protocol pop3
 match protocol smtp
class-map match-any video
 match protocol rtsp
 match protocol vdolive
class-map match-any voice
 match protocol rtp
 match protocol rtcp
 match protocol sip
 match protocol h323
class-map match-any scavenger
 match any
!
!
```

```
policy-map weblinx
 class critical
  set ip dscp ef
  bandwidth percent 5
 class video
  set ip dscp af11
  bandwidth percent 15
 class voice
  set ip dscp af21
  bandwidth percent 15
 class mission_critical
  set ip dscp af31
  bandwidth percent 10
 class web
  set ip dscp af41
  bandwidth percent 5
 class scavenger
  set ip dscp default
!
!
interface FastEthernet0/0
 ip address 192.168.1.1 255.255.255.0
 duplex auto
 speed auto
 service-policy output weblinx
!
interface FastEthernet0/1
```

```
 ip address 192.168.100.2 255.255.255.0
 duplex auto
 speed auto
!
interface Serial0/0/0
 no ip address
 shutdown
 clock rate 2000000
!
interface Serial0/0/1
 no ip address
 shutdown
 clock rate 2000000
!
router eigrp 1
 network 172.16.0.0
 network 192.168.1.0
 network 192.168.100.0
 no auto-summary
!
!
no ip http server
no ip http secure-server
!
!
control-plane
!
```

```
!
line con 0
 exec-timeout 0 0
line aux 0
line vty 0 4
 login
!
scheduler allocate 20000 1000
!
!
End
```

```
CE2#sh run
Building configuration...

Current configuration : 2172 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname CE2
!
boot-start-marker
boot-end-marker
!
!
```

```
no aaa new-model
memory-size iomem 5
!
!
ip cef
!
!
ip host PAGENT-SECURITY-V3 97.32.43.85 87.84.0.0
!
multilink bundle-name authenticated
!
!
voice-card 0
 no dspfarm
!
!
class-map match-any critical
 match protocol bgp
 match protocol ospf
 match protocol eigrp
 match protocol snmp
 match protocol telnet
 match protocol ssh
class-map match-any mission_critical
 match protocol sqlserver
 match protocol sqlnet
class-map match-any web
```

```
 match protocol http
 match protocol secure-http
 match protocol smtp
 match protocol pop3
 match protocol secure-pop3
 match protocol ftp
 match protocol secure-ftp
 match protocol tftp
class-map match-any video
 match protocol rtsp
 match protocol vdolive
class-map match-any voice
 match protocol rtp
 match protocol rtcp
 match protocol sip
 match protocol h323
class-map match-any scavenger
 match any
!
!
policy-map weblinx
 class critical
  bandwidth percent 5
  set ip dscp ef
 class video
  bandwidth percent 15
  set ip dscp af11
```

```
 class voice
  bandwidth percent 15
  set ip dscp af21
 class mission_critical
  bandwidth percent 10
  set ip dscp af31
 class web
  bandwidth percent 5
  set ip dscp af41
 class scavenger
  set ip dscp default
!
!
interface Loopback0
 ip address 172.16.2.1 255.255.255.0
!
interface FastEthernet0/0
 ip address 192.168.2.1 255.255.255.0
 duplex auto
 speed auto
 service-policy output weblinx
!
interface FastEthernet0/1
 ip address 192.168.200.2 255.255.255.0
 duplex auto
 speed auto
 service-policy output weblinx
```

```
!
interface Serial0/0/0
 no ip address
 shutdown
 no fair-queue
 clock rate 2000000
!
interface Serial0/0/1
 no ip address
 shutdown
 clock rate 2000000
!
router eigrp 1
 network 172.16.0.0
 network 192.168.2.0
 network 192.168.200.0
 no auto-summary
!
!
!
ip http server
no ip http secure-server
!
control-plane
!
!
!
```

```
line con 0
 exec-timeout 0 0
line aux 0
line vty 0 4
 login
!
scheduler allocate 20000 1000
!
!
End
```

9.2 Appendix B

Routing and VRF tables

P1#sh ip route

```
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

     10.0.0.0/8 is variably subnetted, 7 subnets, 2 masks
C       10.0.2.0/24 is directly connected, FastEthernet0/1
O       10.0.3.0/24 [110/2] via 10.0.1.2, 05:42:55, FastEthernet0/0
C       10.0.1.0/24 is directly connected, FastEthernet0/0
```

O 10.0.104.1/32 [110/3] via 10.0.1.2, 05:42:55, FastEthernet0/0

O 10.0.103.1/32 [110/2] via 10.0.2.3, 05:42:55, FastEthernet0/1

O 10.0.102.1/32 [110/2] via 10.0.1.2, 05:42:55, FastEthernet0/0

C 10.0.101.1/32 is directly connected, Loopback0

P2#sh ip route

Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP

D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area

N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2

E1 - OSPF external type 1, E2 - OSPF external type 2

i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2

ia - IS-IS inter area, * - candidate default, U - per-user static route

o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

10.0.0.0/8 is variably subnetted, 7 subnets, 2 masks

O 10.0.2.0/24 [110/2] via 10.0.1.1, 05:48:54, FastEthernet0/0

C 10.0.3.0/24 is directly connected, FastEthernet0/1

C 10.0.1.0/24 is directly connected, FastEthernet0/0

O 10.0.104.1/32 [110/2] via 10.0.3.4, 05:48:54, FastEthernet0/1

O 10.0.103.1/32 [110/3] via 10.0.1.1, 05:48:54, FastEthernet0/0

C 10.0.102.1/32 is directly connected, Loopback0

O 10.0.101.1/32 [110/2] via 10.0.1.1, 05:48:54, FastEthernet0/0

PE1#sh ip route

Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP

D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area

N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2

E1 - OSPF external type 1, E2 - OSPF external type 2

i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2

ia - IS-IS inter area, * - candidate default, U - per-user static route

o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

10.0.0.0/8 is variably subnetted, 7 subnets, 2 masks

C 10.0.2.0/24 is directly connected, FastEthernet0/1

O 10.0.3.0/24 [110/3] via 10.0.2.1, 05:46:08, FastEthernet0/1

O 10.0.1.0/24 [110/2] via 10.0.2.1, 05:46:08, FastEthernet0/1

O 10.0.104.1/32 [110/4] via 10.0.2.1, 05:46:08, FastEthernet0/1

C 10.0.103.1/32 is directly connected, Loopback0

O 10.0.102.1/32 [110/3] via 10.0.2.1, 05:46:08, FastEthernet0/1

O 10.0.101.1/32 [110/2] via 10.0.2.1, 05:46:08, FastEthernet0/1

PE1#sh ip route vrf weblinx

Routing Table: weblinx

Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP

D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area

N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2

E1 - OSPF external type 1, E2 - OSPF external type 2

i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2

ia - IS-IS inter area, * - candidate default, U - per-user static route

o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

172.16.0.0/24 is subnetted, 1 subnets

B 172.16.2.0 [200/156160] via 10.0.104.1, 05:45:23

B 192.168.200.0/24 [200/30720] via 10.0.104.1, 02:44:35

C 192.168.1.0/24 is directly connected, FastEthernet0/0

B 192.168.2.0/24 [200/0] via 10.0.104.1, 05:45:23

D 192.168.100.0/24 [90/30720] via 192.168.1.1, 00:38:06, FastEthernet0/0

PE2#sh ip route

Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP

D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area

N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2

E1 - OSPF external type 1, E2 - OSPF external type 2

i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2

ia - IS-IS inter area, * - candidate default, U - per-user static route

o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

10.0.0.0/8 is variably subnetted, 7 subnets, 2 masks

O 10.0.2.0/24 [110/3] via 10.0.3.2, 05:52:12, FastEthernet0/1

C 10.0.3.0/24 is directly connected, FastEthernet0/1

O 10.0.1.0/24 [110/2] via 10.0.3.2, 05:52:12, FastEthernet0/1

C 10.0.104.1/32 is directly connected, Loopback0

O 10.0.103.1/32 [110/4] via 10.0.3.2, 05:52:12, FastEthernet0/1

O 10.0.102.1/32 [110/2] via 10.0.3.2, 05:52:12, FastEthernet0/1

O 10.0.101.1/32 [110/3] via 10.0.3.2, 05:52:12, FastEthernet0/1

PE2#sh ip route vrf weblinx

Routing Table: weblinx

Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP

D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area

N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2

E1 - OSPF external type 1, E2 - OSPF external type 2

i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2

ia - IS-IS inter area, * - candidate default, U - per-user static route

o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

172.16.0.0/24 is subnetted, 1 subnets

D 172.16.2.0 [90/156160] via 192.168.2.1, 05:56:35, FastEthernet0/0

D 192.168.200.0/24 [90/30720] via 192.168.2.1, 02:50:40, FastEthernet0/0

B 192.168.1.0/24 [200/0] via 10.0.103.1, 05:51:09

C 192.168.2.0/24 is directly connected, FastEthernet0/0

B 192.168.100.0/24 [200/30720] via 10.0.103.1, 00:43:59

CE1#sh ip route

Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP

D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area

N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2

E1 - OSPF external type 1, E2 - OSPF external type 2

i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2

ia - IS-IS inter area, * - candidate default, U - per-user static route

o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

172.16.0.0/24 is subnetted, 1 subnets

D 172.16.2.0 [90/158720] via 192.168.1.3, 00:48:43, FastEthernet0/0

D 192.168.200.0/24 [90/33280] via 192.168.1.3, 00:48:43, FastEthernet0/0

C 192.168.1.0/24 is directly connected, FastEthernet0/0

D 192.168.2.0/24 [90/30720] via 192.168.1.3, 00:48:43, FastEthernet0/0

C 192.168.100.0/24 is directly connected, FastEthernet0/1

CE2#sh ip route

Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP

D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area

N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2

E1 - OSPF external type 1, E2 - OSPF external type 2

i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2

```
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

    172.16.0.0/24 is subnetted, 1 subnets
C       172.16.2.0 is directly connected, Loopback0
C    192.168.200.0/24 is directly connected, FastEthernet0/1
D    192.168.1.0/24 [90/30720] via 192.168.2.4, 05:53:55, FastEthernet0/0
C    192.168.2.0/24 is directly connected, FastEthernet0/0
D    192.168.100.0/24 [90/33280] via 192.168.2.4, 00:38:43, FastEthernet0/0
```