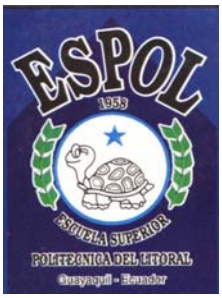


ESCUELA SUPERIOR POLITECNICA DEL LITORAL

Facultad de Ingeniería en Electricidad y Computación

"DISEÑO E IMPLEMENTACION MEDIANTE EL SIMULADOR DYNAMIPS DE UNA RED MPLS PARA LA CONEXIÓN WAN DE UNA EMPRESA MEDIANA CON SUS SUCURSALES"

TESIS DE GRADO

Previo a la obtención del Título de:

INGENIERO EN ELECTRONICA Y TELECOMUNICACIONES

REALIZADO POR:

ALEX VIDAL BALLESTEROS GRACIA

ANDRES ALEXANDER CHIRIBOGA JARRIN

LUIS MIGUEL VILLEGAS GAVILANES

DIRECTOR: MSC. IVONNE MARTIN MORENO

GUAYAQUIL – ECUADOR

2007

DEDICATORIA

Dedico el presente trabajo a mis queridos padres Haydeé Gracia y Vidal Ballesteros, personas de bien que son muestra de trabajo, honestidad y honradez, que depositaron su confianza en mí; con mucho cariño, amor, comprensión y dedicación, me supieron inculcar valores dejándome como mejor herencia el tesoro mas valioso que a un ser humano le pueden dejar como lo es la preparación.

A mis hermanas Verónica y Karen, quienes también luchan en la vida quedando yo, como ejemplo para ellas y así nunca pierdan las ganas de salir adelante. A mis tíos Margarita e Iván quienes se preocuparon al igual que mis padres durante mis años de carrera universitaria enseñándome y haciéndome entender que la preparación da muestras y fé, de sabiduría y lucha constante en el diario vivir.

A la memoria de mis abuelitos paternos Mercedes y Francisco, a mis abuelitos maternos Macario e Isabel, a todos primos y demás mis familiares que siempre me estuvieron apoyando en los buenos y en los malos momentos y supieron enseñarme que en la vida no se fracasa mientras no se deje de luchar constantemente.

Alex Vidal Ballesteros Gracia.

Dedico este valioso trabajo con todo cariño a mis queridos padres, Jorge Alfredo Chiriboga Sigüenza y Rosa Elizabeth Jarrín Castro, quienes durante toda mi vida estudiantil me han ayudado a lograr la superación personal brindándome invalorables consejos que han hecho de mi, una persona de bien en todos los aspectos.

A mi tía Jovita Chiriboga Sigüenza, que en paz descanse, quien confió en mí, dándome su apoyo incondicional en mi etapa de escuela y que poco a poco fue formando en mí, valores imprescindibles en mi desarrollo como persona.

A mis queridos hermanos, Jorge y Vanessa, quienes me ofrecieron sus consejos para que no desmaye en mi afán de ser un profesional y lograr alcanzar numerosas metas que me propuse en la vida.

A toda mi familia que al pasar del tiempo me ha ayudado de una u otra forma en lograr los objetivos trazados y que me servirán de mucha ayuda para la prosperidad en el futuro.

Andrés Alexander Chiriboga Jarrín.

Dedico mi tesis con todo mi cariño y amor al Dios Todopoderoso ya que, gracias a el tengo la vida, el conocimiento y la valentía de seguir adelante.

A mi madre Concepción Gavilanes y a la memoria de mi padre Jorge Villegas que con amor y humildad supieron encaminarme por el sendero del saber inculcándome desde mis inicios educativos apreciar y valorar sus esfuerzos para que sea una persona de bien.

A mis hermanos Paulina y Jorge que con su ayuda y compañía, nos superamos día a día consiguiendo juntos el éxito deseado.

A mi tío Víctor que ha sido como un padre y que gracias a su apoyo y consejos estoy logrando la meta propuesta de la vida profesional.

A mis desaparecidos abuelos Luciano , Enriqueta y Víctor, a mi abuela Rosa que todavía me acompaña y al resto de familiares que con sus muestras de afecto y con sus granos de arena confían en mis estudios y han hecho posible mi superación.

Luis Miguel Villegas Gavilanes.

AGRADECIMIENTO

Como ser supremo y Rey Celestial, agradecemos a Dios por habernos dado vida para poder luchar sin pensar en obstáculos y seguir adelante enfrentando lo difícil y fuerte de manera inteligente.

Nuestro profundo agradecimiento a nuestra querida directora de Tesis, Ing. Ivonne Martín Moreno, por la confianza que nos brinda y por su constante ayuda en el momento de desarrollar un valioso trabajo que impulsa a la sociedad del conocimiento y abre las puertas a que muchas personas puedan desarrollar y madurar nuevas ideas a través de la investigación.

Al Ingeniero Servio Lima, "Gerente Técnico de Telconet", quien nos abrió las puertas de la empresa para conocer mas en lo referente a las tecnologías que soporta un Carrier de hoy en día.

Al Ingeniero Juan Carlos Gaibor, por haber ofrecido su ayuda incondicional y por sus valiosas enseñanzas impartidas durante un periodo que nos sirvió muchísimo para reforzar nuestras ideas y adquirir nuevos conocimientos.

MOTIVACIÓN

La motivación que lleva a desarrollar el presente trabajo esta inspirada en el enorme crecimiento de las redes de datos en la actualidad y la tendencia a la migración de nuevas tecnologías que se están desarrollando para luego estandarizarse y lograr así sentar bases de los que serán las tecnologías del presente y futuro que pueden ser estudiadas y reforzadas mediante una investigación profunda analizando el punto de vista de los antecedentes para poder establecer grandes análisis de consecuencias que puedan darse a la hora de implementar un backbone con nuevas tecnologías de transmisión de datos.

DECLARACION

"La responsabilidad por los hechos, ideas y doctrinas expuestas en este trabajo nos corresponden exclusivamente; y el patrimonio intelectual de la misma a la ESCUELA SUPERIOR POLITECNICA DEL LITORAL"

Alex Vidal Ballesteros Gracia

Andrés Alexander Chiriboga Jarrín

Luis Miguel Villegas Gavilanes

TRIBUNAL DE GRADO

Ing. Ivonne Martín

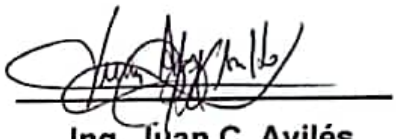
Directora

Ing. Hólger Cevallos

Presidente

Ing. Albert Espinal

Vocal Principal 1



Ing. Juan C. Avilés

Vocal Principal 2

INDICE GENERAL

ÍNDICE DE TABLAS

ÍNDICE DE FIGURAS

RESUMEN

El presente trabajo de Tesis se lo ha desarrollado pensando en los servicios que las nuevas tecnologías ofrecerán y además pensando hacia donde las tecnologías actuales tienden a migrar, es por ello, que el desarrollo de esta Tesis tiene objetivos claves como lo son:

- Dar a conocer los conceptos básicos de la tecnología MPLS que se aplicarán y se validarán en este proyecto de tesis.

- Dar a conocer el funcionamiento y uso del simulador *Dynamips*, que se emplea en ésta tesis y validar el diseño a presentarse utilizando el mencionado simulador bajo entorno del sistema operativo *Linux*

- Presentar soluciones de proveedores de servicio a empresas que deseen entablar redes de voz y datos con sus sucursales mediante el transporte confiable punto a punto utilizando MPLS.

- Validar el desempeño de la tecnología MPLS mediante aplicaciones como: Redes Privadas Virtuales, Ingeniería de Tráfico y Calidad de Servicio.

- Realizar comparaciones en lo que corresponde a la tecnología MPLS con las tecnologías actuales como IP, en las redes de datos.

Este trabajo está estructurado en cinco capítulos en los cuales precisamente se busca que la tecnología MPLS sea entendida excelentemente por aquellos que deseen aprender y continuar realizando sus investigaciones en lo que a las tecnologías futuras respecta.

El primer capítulo esta dedicado a dar una teoría introductoria a los principios de la tecnología MPLS, los objetivos de la misma, su viabilidad, y dando a conocer algunos términos de interés que se irán conociendo detalle a detalle en capítulos posteriores.

El segundo capítulo se centra en la descripción funcional de una red MPLS indicando las funcionalidades de los elementos participantes en un dominio, induciendo además una teoría introductoria a las diversas aplicaciones que se pueden ofrecer al tener a esta tecnología ya implementada en el *Backbone* de un proveedor de servicios de red.

El tercer capítulo se basa en lo referente al diseño que puede ser ofrecido por un proveedor de servicios a una empresa que desee conectar sus sucursales por medio de tecnologías de transporte que le garanticen fiabilidad en el manejo y envío de su información. Se da también a conocer la topología de red y las herramientas representativas para el diseño de la misma.

El cuarto capítulo está dedicado a la implementación del diseño de la topología de la red MPLS con el uso de herramientas muy poderosas tales como el simulador *Dynamips* el cual basa su funcionamiento bajo entorno *Linux,* y del cual se detalla además su funcionamiento. En este capítulo también se presenta los mecanismos de configuración de los elementos participantes en el diseño de la red y la funcionalidad de los mismos.

En el capítulo cinco se detalla un análisis de los costos que demanda la implementación de una red MPLS y los enlaces de los clientes.

INTRODUCCIÓN

Durante los últimos años ha existido un enorme crecimiento en lo que concierne al uso de las tecnologías de redes y transmisión de datos que están al alcance de los proveedores. En la actualidad ya es posible contar con un servicio rápido, estable, eficiente y multifuncional sobre el cual pueden correr muchas aplicaciones (voz, datos, video) a un precio al alcance de usuarios que se expanden con el pasar de los días. Uno de los propósitos que siempre se ha buscado es conseguir la mejora en la calidad de los servicios que un proveedor le puede brindar a las empresas por medio del uso de las redes de datos en el mundo. Hoy en día las tecnologías existentes como IP están diseñadas para que brinden seguridad y sean capaces de reestablecer conectividad luego de que se presenten fallas en algún elemento de red. Aunque la conectividad pueda ser reestablecida, el tiempo que esto demande podría no estar en el límite para lo aceptable en lo que respecta a servicios de alta prioridad. Por esta razón se estudia las posibilidades para que un proveedor de servicio implemente en sus redes sistemas confiables que puedan dar a los clientes seguridad al momento de conectar sus redes, adaptando el protocolo IP a las tecnologías de WAN que hoy en día se pueden utilizar.

1 GENERALIDADES

1.1 Principios de la tecnología MPLS

Puede decirse que TCP/IP hoy en día es una solución clásica y estándar al transporte de información en las redes, ya que el mismo ha sido aceptado ampliamente por todas las comunidades, y ha sido hasta nuestros días una solución aceptable para el envío de información, utilizando encaminamiento de paquetes ofreciendo muchas garantías de entrega.

1.1.1 La Tecnología Actual TCP/IP

El Protocolo de Control de Transporte / Protocolo de Internet (TCP/IP), es un conjunto de reglas desarrolladas para permitir que los computadores que cooperan entre sí puedan compartir recursos a través de una red. TCP/IP consta de 4 capas según el modelo de referencia creado por el Departamento de Defensa de los EEUU que fue diseñado con la finalidad de que una red pudiera sobrevivir ante cualquier circunstancia. El modelo de protocolos TCP/IP ayudó a solucionar los inconvenientes, los cuales eran necesarios evitar al momento de transmitir los datos desde una fuente hacia destino a través una infraestructura confiable.

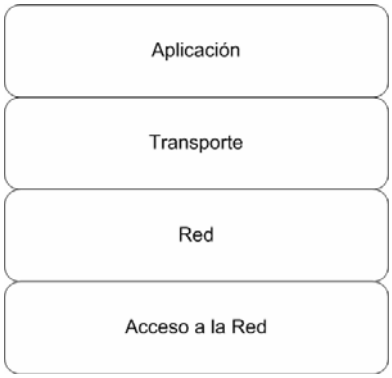


Figura 1-1.- Modelo de Capas TCP / IP

Como se puede observar, las diferentes capas del modelo TCP / IP (fig. 1-1), es importante también describir las funciones que realizan las mismas para comprender el manejo de la información con esta tecnología.

Capa de Aplicación: Esta Capa se encarga del manejo de los protocolos de alto nivel, aspectos de representación, codificación y control de dialogo.

Capa de Transporte: Como su nombre lo indica, aquí se proporciona servicios de transporte desde una fuente hacia un destino. La capa de transporte puede verse como una conexión lógica entre los puntos finales de una red (Emisor, Receptor). Cabe recalcar que los protocolos de transporte segmentan y reensamblan los datos transmitidos por las capas superiores en un mismo flujo de datos, o conexión lógica entre los extremos.

Además se debe conocer que el control de punta a punta se proporciona con técnicas de ventanas deslizantes; y la confiabilidad de los números de secuencia y acuses de recibo es el deber básico de la capa de transporte cuando la misma emplea TCP.
La gráfica (fig. 1-2) muestra el formato de un segmento TCP. Los campos contenidos en este segmento presentan la manera mas precisa de describir la confiabilidad de TCP.

<table>
<tr><td colspan="2">Bit 0</td><td colspan="2">Bit 15</td><td colspan="2">Bit 16</td><td>Bit 31</td></tr>
<tr><td colspan="4">Puerto origen (16)</td><td colspan="3">Puerto destino (16)</td></tr>
<tr><td colspan="7">Número de secuencia (32)</td></tr>
<tr><td colspan="7">Número de acuse de recibo (32)</td></tr>
<tr><td colspan="2">Longitud del encabezado (4)</td><td colspan="2">Reservado (6)</td><td>Bits de código (6)</td><td colspan="2">Ventana (16)</td></tr>
<tr><td colspan="4">Cheksum (16)</td><td colspan="3">Urgente (16)</td></tr>
<tr><td colspan="7">Opciones (0 ó 32 si las hay)</td></tr>
<tr><td colspan="7">Datos (varía)</td></tr>
</table>

Figura 1-2.- Segmento TCP

A más de TCP existe una nueva opción como protocolo que trabaja en capa de transporte como los es, el muy conocido UDP *(User Datagram Protocol)* sobre el cual pueden trabajar muchas aplicaciones. A diferencia de TCP, UDP no es orientado a conexión ni ofrece confiabilidad en la transmisión de paquetes ya que no emplea métodos como ventanas deslizantes ni acuses de recibo de modo que, son los protocolos de capas superiores (Aplicación) los que deben brindar la detección de errores. Al igual que TCP, UDP cuenta con su Formato (fig. 1-3)

<table>
<tr><td colspan="2">BIT 0</td><td colspan="2">BIT 15 BIT 16</td><td>BIT 31</td></tr>
<tr><td colspan="2">Puerto Origen (16)</td><td colspan="3">Puerto Destino (16)</td></tr>
<tr><td colspan="2">Longitud (16)</td><td colspan="3">Checksum (16)</td></tr>
<tr><td colspan="5">Datos (de haber alguno)</td></tr>
</table>

Figura 1-3.- Segmento UDP

Tanto TCP como UDP tienen como protocolo subyacente de capa de red a IP, y por esta sencilla razón, para el transporte a través de la red, estos protocolos se encapsulan en el paquete IP.

Capa de Red o Internet: El propósito en esta capa de la arquitectura TCP/IP es encapsular los segmentos de datos de capas superiores en paquetes y seleccionar la mejor ruta para enviarlos a través de una red.

El protocolo IP es la parte central del conjunto de protocolos de la tecnología TCP/IP. IP es un protocolo de Capa de Red el cual ofrece un servicio de envío de paquetes no orientado a conexión, siendo además un protocolo orientado a Datagramas que trata a cada paquete de manera independiente, de modo que cada paquete deberá obtener la información necesaria para ser direccionado de manera correcta. IP no tiene garantía de entrega de paquetes ni garantía de integridad en la información recibida ya que no comprueba el contenido de paquetes ni posee mecanismos de confirmación para determinar si el paquete ha alcanzado su destino.

En cuanto a las capas sobre las que se trabajar el protocolo IP se puede encontrar diversas posibilidades, y como es de conocimiento que este protocolo oculta la tecnología subyacente a sus usuarios, el mismo puede verse ejecutado sobre cualquier medio de transporte (ATM, Frame Relay, PPP, etc.).

La gráfica (fig. 1-4) muestra la estructura del paquete IP con sus respectivos campos, los mismos que son detallados por lo cual encaminan al lector a obtener un concepto claro de cómo son empaquetados los segmentos de datos que vienen desde la capa superior TCP.

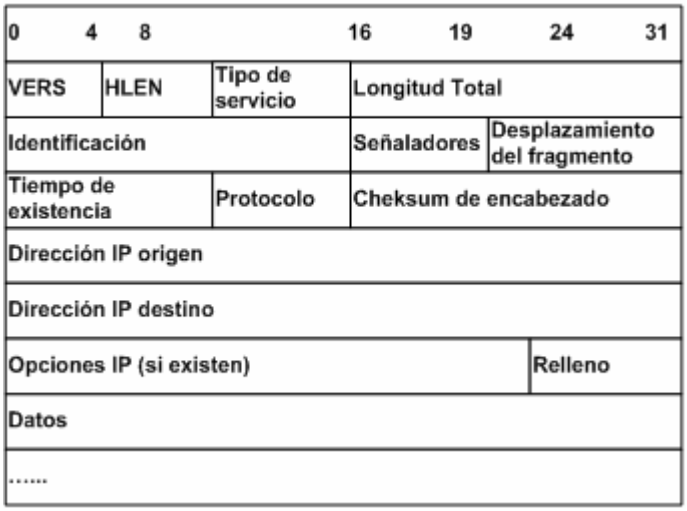


Figura 1-4.- Paquete IP

- **Versión:** Identifica la versión del protocolo (IPv4, IPv6)

- **HLEN o IHL *(Internet Header Length):*** Longitud de la cabecera de Internet. 32 bits.

- **ToS *(Type of Service)*:** Indica la prioridad y el tipo de transporte que se desea utilizar.

- **Longitud Total:** Especifica el tamaño del datagrama en octetos. Este campo puede ser hasta 65535 octetos.

- **Identificación:** Un valor asignado por la capa de transporte para permitir el ensamblado de los fragmentos.

- **Señaladores:** Controlan las opciones de la fragmentación.

- **Desplazamiento del Fragmento:** Indica el lugar donde se encuentra este fragmento en el datagrama original.

- **TTL (Tiempo de Existencia):** Indica el mayor número de saltos que el datagrama puede realizar a lo largo de la red.

- **Protocolo:** Indica el protocolo de Capa Superior (TCP o UDP).

- **Checksum de Encabezado:** Suma de Chequeo de 16 bits de la cabecera del datagrama.

- **Dirección Origen / Destino:** Valores de 32 bits que indican el emisor y el receptor del paquete.

- **Opciones:** Información para el control de la red, enrutamiento y gestión.

- **Relleno:** Campo de relleno para adaptar la longitud de la cabecera a 20 *bytes*.

- **Datos:** Los datos de la Capa de transporte.

Capa de Enlace o Acceso a la Red: También denominada capa de usuario a red. La capa de acceso de red es aquella que maneja todos los aspectos que un paquete IP requiere para efectuar un enlace físico real con los medios de la red.

En esta capa se incluyen detalles de las tecnologías tanto de LAN como de WAN. Entre las funciones principales de la capa de Acceso de Red se incluyen la asignación de direcciones IP a direcciones físicas de acceso al medio (Media Access Control – MAC) y el encapsulamiento de los paquetes IP en tramas.

Al conocerse y detallarse cada una de las capas de TCP/IP, es de importancia detallar el establecimiento de comunicación entre dos sistemas o dos puntos. La base de la comunicación en TCP/IP cae en el direccionamiento o enrutamiento IP, para lo cual se debe tener en cuenta el tipo de direcciones a usar y cómo estas se pueden clasificar

En TCP / IP para establecer una sesión y comunicar dos equipos finales, es necesaria una dirección IP en cada equipo y por esto se debe entender el concepto de una dirección IP. Una dirección IP es una secuencia de unos y ceros limitado por 32 bits (para el caso de IPv4), separado por puntos en cuatro octetos (**ej.** 192.168.3.1). Una dirección comienza con un número de red empleado por el enrutamiento, seguido de una dirección local para la red interna.

Las direcciones IP de la versión 4, se dividen en cuatro clases basadas en función del tamaño que corresponda a la parte de red en la dirección IP:

Clase A: Para esta clase, el *bit* de orden mas alto es “0”, los siguientes 7 *bits* del primer octeto son para la red y los tres octetos restantes son para la dirección local.

Clase B: Esta clase puede ser diferenciada de la clase anterior dado que, los dos *bits* de orden más alto son “10”, los siguientes 14 *bits* de los dos primeros octetos son para la red y los otros dos octetos restantes son para la dirección local.

Clase C: A diferencia de la clase B, los tres *bits* de orden mas alto son “110”, los siguientes 21 *bits* de los tres primeros octetos son para la red y el último octeto corresponde a la dirección local.

Clase D: Esta clase está orientada en lo que se relaciona con pruebas de *multicasting* y para ello, los cuatro *bits* de orden más alto son “1110”, seguidos de la dirección *multicast*.

Como se menciona, así como IP permite *Multicasting*, también soporta *Broadcasting*.

En el direccionamiento IP existen rangos de direcciones que pueden ser utilizadas para direccionamiento público y direccionamiento privado según especifica la Agencia de Asignación de Números de Internet (IANA).

Las direcciones IP públicas son muy utilizadas en lo que va de la mano con el uso de los dominios públicos como el Internet, y son muy exclusivas ya que dos equipos de redes que se conectan a una red pública nunca pueden tener la misma dirección de IP dado que las direcciones públicas son globales y están estandarizadas. Por esta razón el IANA especifica lo siguiente: “Todos los equipos de Redes que se conectan a Internet acuerdan adaptarse al sistema”.

Las direcciones públicas deben ser obtenidas de un ISP o un registro, a un costo.

Por el crecimiento imparable del Internet, las direcciones IP públicas han comenzado a escasearse. En la actualidad se han desarrollado nuevos esquemas de direccionamiento, tales como el enrutamiento entre dominios sin clase y el IPv6, para ayudar a encontrarle una solución a este problema. Otra de las soluciones al problema de agotamiento en el espacio de direcciones públicas son las direcciones IP privadas. Para este esquema de direccionamiento las redes que nos están conectadas a la *Internet* pueden utilizar cualquier dirección local o de usuario, siempre y cuando cada usuario

dentro de la red privada sea exclusivo. Existe una gran cantidad de redes privadas junto con las redes públicas. Sin embargo no se recomienda que una red privada utilice una dirección cualquiera dado que, con el tiempo, la red podría conectarse a Internet. El *RFC 1918* asigna tres bloques para uso interno y privado. Estos 3 bloques están basados en un rango de direcciones de clase A, un rango de direcciones de clase B, y un rango de direcciones de clase C. Si alguna dirección se encuentra dentro de estos rangos, entonces ella no será enrutada hacia el Backbone de la *Internet*, sin embargo, la conexión de una red que utiliza direcciones privadas a la *Internet* requiere que las direcciones privadas se conviertan a públicas. Este proceso se conoce como "Traducción de direcciones de red *(Network Address Traslation – NAT)*". Para llegar a un entendimiento general es necesario saber que un ruteador es el dispositivo que realiza la *NAT*.

Clase	***Rango de Direcciones***
A	10.0.0.0 – 10.255.255.255
B	172.16.0.0 – 172.31.255.255
C	192.168.0.0 – 192.168.255.255

Tabla 1-1.- Rango de Direcciones IP Privadas

Como se puede observar (tabla 1-1) los rangos para cada clase están definidos y las direcciones IP que se encuentran fuera de estos límites son utilizadas para direccionamiento público.

1.1.2 Presentación de una nueva arquitectura.

En la actualidad los avances en *Hardware* y la nueva visión que se tiene a la hora de manejar las redes, está dando lugar al empleo de tecnologías de conmutación de etiquetas añadidas a paquetes IP *(Label Switching)* que

aportan con velocidad, calidad de servicio y facilitan la gestión de los recursos de una red.

Lo que se busca es romper con los paradigmas del enrutamiento, el cual está muy extendido en entornos empresariales, académicos, etc. Hoy en día, en el enrutamiento convencional IP, como es de conocimiento para muchos se vuelve rutinario ya que, cada ruteador en una red toma sus decisiones de encaminamiento para cada paquete entrante, y lo hace basándose en su tabla de enrutamiento para encontrar el siguiente salto para el paquete. Dichas tablas son creadas estáticamente o mediante protocolos dinámicos como pueden ser BGP, OSPF, RIP, IS – IS, IGRP, EIGRP, entre otros.
Aunque está ampliamente desarrollado, IP tiene ciertas restricciones que se han dado a conocer a lo largo del tiempo, lo cual ha disminuido la flexibilidad del método de envío de paquetes. Nuevas técnicas son requeridas para direccionar y expandir la funcionalidad de una infraestructura de red basada en IP. Actualmente, para entender los inconvenientes que afectan la escalabilidad y la flexibilidad de la redes con envío tradicional, se debe comenzar con una revisión de algunos mecanismos de envío básico de IP y su interacción con el refuerzo de una infraestructura.

Como es de conocer, el rediseño de una red para encaminarla hacia las nuevas tendencias supone un enorme gasto de tiempo y dinero, es por esa razón que se debe siempre analizar desde el punto de vista de las tecnologías existentes al momento de migrar una red hacia las tendencias del mañana. Esto se lo hace con la finalidad de encontrar los campos comunes y las diferencias fundamentales, así como se han encontrado las primeras soluciones al momento en que se integra un servicio de red.

Una solución para hoy y para el mañana a los problemas del envió tradicional, es *Multiprotocol Label Switching – MPLS*, lo cual es un conjunto

de procedimientos que combinan el desempeño, calidad de servicio y gestión del tráfico de los paradigmas de barrido de etiquetas en capa de acceso a la red con la escalabilidad y flexibilidad de funcionalidades del enrutamiento en la capa de red o *Internet*.

Entre los principios de MPLS se cuenta el asignar etiquetas de longitud corta y fija a paquetes en la frontera a un dominio MPLS, y entonces utilizar estas etiquetas en lugar de las cabeceras de los paquetes IP para enviar dichos paquetes a través de los caminos preestablecidos dentro de una red MPLS.

En MPLS, la ruta por la que es enviado el paquete a través del dominio es asignada una sola vez a medida que un paquete ingresa a la red. Los nodos a lo largo de una ruta no toman decisiones de envío para un paquete específico, ellos utilizan una etiqueta del paquete como índice en una tabla que les especifica el siguiente salto del paquete. Antes que un ruteador envíe el paquete, éste cambia la etiqueta en el mismo a una que es utilizada para enviar datos por el siguiente ruteador en el camino. Este es el concepto arquitectural principal del reforzamiento MPLS; la separación del plano de control del plano de envío en los elementos de conmutación.

1.1.3 Viabilidad Técnica y Operacional de MPLS

Para que un proveedor utilice las nuevas tendencias en lo que trata al uso de tecnologías de transmisión de datos y banda ancha, debe analizar si la migración de una red a otra más actual es viable, tanto técnicamente como operacionalmente.

1.1.3.1 Viabilidad Técnica

La arquitectura MPLS necesita de dos elementos de red propios: Un equipo que realiza las funciones de conmutar etiquetas, y otro equipo que trabaja en dos entornos de tecnologías diferentes.

Para utilizar estos elementos es necesario sustituir el *software* utilizado por los equipos actuales en una red para que tengan compatibilidad con el envío MPLS. Para el caso de un equipo del núcleo de la red es necesario que las funcionalidades se basen únicamente en la conmutación e intercambio de etiquetas. Por otra parte, en un dispositivo de frontera de red se debe contar con funcionalidades tanto de envío tradicional IP, como de envío MPLS.

Dado que muchas de las modificaciones solamente se dan a nivel de sistema operativo, la solución MPLS es viable técnicamente facilitando el uso de mismos dispositivos y de las plataformas específicas, a los cuales se les ha incrementado sus funcionalidades.

1.1.3.2 Viabilidad Operacional

Para obtener de MPLS una solución viable operacionalmente es necesario añadir características tanto de envío como de control a los dispositivos de red para que ellos realicen las tareas compatibles con la nueva tecnología.

En la componente de control de un dispositivo de núcleo de red se añaden funcionalidades que permitan entablar y relacionar conceptos de enrutamiento IP tradicional con la conmutación de etiquetas. Por otro lado, para los dispositivos de las fronteras es necesario añadir herramientas que

permitan trabajar bajo dos entornos entablando el envío tradicional con el envió MPLS.

1.2 MPLS – Multiprotocol Label Switching

MPLS combina flexibilidad y fiabilidad en las comunicaciones ofreciendo niveles de rendimiento diferenciados y priorización del tráfico de red, así como diversas aplicaciones en lo que a conexión entre puntos se conoce.

1.2.1 Objetivos de la tecnología MPLS

La tecnología MPLS centra su objetivo principal en estandarizarse para servir de base en lo que es la combinación de la conmutación de paquetes y el enrutamiento tradicional, para lo cual es necesario integrar la parte MPLS en el plano de control de la capa de red. Por ello se ha desarrollado MPLS con la finalidad de satisfacer requerimientos como:

- Ejecutarse sobre cualquier tecnología, en la capa de Red.

- Soportar flujo de tráfico tanto *Unicast* como *Multicast*.

- Ser escalable, para soportar el crecimiento de las estructuras de redes corporativas y la expansión de la *Internet*.

- Ser compatible con el modelo de servicios integrados del IETF, incluyendo el protocolo RSVP.

- Ser compatible con los procedimientos de operación, administración y mantenimiento de las actuales redes IP.

1.2.2 Elementos participantes en una Red MPLS

Una red consta de varios elementos y términos los cuales describen su arquitectura. Para MPLS existen una variedad de elementos que permiten el acceso a la red y elementos que brindan servicios confiables en el núcleo de la misma, y son los que definen como se hizo hincapié, la estructura de un Dominio MPLS.

1.2.2.1 Label Switch Router (LSR) – (Router Conmutador de Etiquetas)

El LSR como su nombre lo indica es un ruteador – conmutador de etiquetas que basa su funcionamiento de envío en el chequeo de la etiqueta o pila de etiquetas que ha sido añadida a un paquete IP en la frontera de ingreso al dominio MPLS. No realiza funciones de chequeo de capa de red ya que para el envío basta con analizar la etiqueta contenida en el paquete IP etiquetado, la cual le indica su siguiente salto. El LSR remueve la etiqueta y asigna otra para indicar el siguiente salto dentro de la red MPLS.
En su estructura interna el LSR consta de un Plano de Control y un Plano de Datos o Plano de Envío los mismos que se muestran en la figura 1-5.

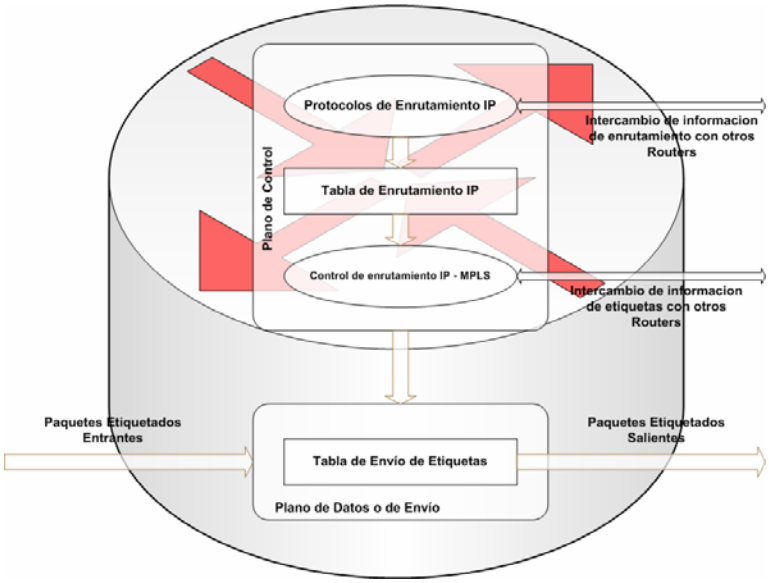


Figura 1-5.- Diagrama de Bloques del LSR

1.2.2.2 Label Edge Router (LER) o Edge – LSR - (Router de Frontera de Etiquetas)

El LER o *Edge* – LSR es un ruteador el cual se encuentra en la frontera de una red MPLS. Se encarga de realizar y brindar funcionalidades concernientes a lo que a aplicaciones del cliente se refiere cuando éste esta conectado a la red de un proveedor MPLS. Este elemento se encuentra presente tanto en el ingreso como en el egreso de la red, cumpliendo las funciones principales de la misma. Estos ruteadores cumplen funciones ya sea para un Dominio MPLS como para un Dominio no MPLS.

En el ingreso a la red MPLS ellos cumplen con la función de chequear un paquete IP entrante, al mismo que luego le asigna una etiqueta o pila de etiquetas obtenida bajo ciertas técnicas las mismas que se presentan en secciones posteriores. Al asignarle la etiqueta o pila de etiquetas al paquete IP, el LER se encarga de enviar el paquete etiquetado a los LSRs que se

encuentran en el núcleo de la red MPLS. En adición al LSR este ruteador si realiza las funciones de un ruteador normal en la capa de red, es decir, el chequeo correspondiente de la cabecera IP del paquete para asignar la etiqueta según la FEC correspondiente a su dirección IP.

Al encontrarse en la frontera de salida del Dominio MPLS, el *Edge – LSR* cumple con la función de chequear el paquete etiquetado MPLS, y al realizar el chequeo de etiquetas correspondiente toma la decisión de enviar el paquete a su siguiente salto verificando que el mismo sea un Dominio MPLS o IP.

- Si el siguiente salto es otro dominio MPLS, el LER se encarga de remover la etiqueta y asignar otra que indique el siguiente salto MPLS para enviar el paquete a su destino basándose en la etiqueta que ha obtenido de su ruteador de *downstream*.

- Si el siguiente salto es un dominio no MPLS, el LER se encarga de remover la etiqueta y realizar un chequeo en su tabla de enrutamiento para lo cual busca el siguiente salto que debe ser alcanzado para que el paquete que vuelve a su estado IP original llegue a su destino.

1.2.2.3 Label (Etiqueta)

La etiqueta en MPLS es un identificador de longitud corta y constante que se emplea para identificar una clase de envío equivalente (FEC), normalmente con carácter local. La etiqueta o pilas de etiquetas indican el camino que un paquete IP etiquetado debe seguir hasta alcanzar su destino. En la tecnología TCP/IP la etiqueta se encuentra empaquetada en la cabecera MPLS.

1.2.2.4 Label Switched Path (LSP) – (Ruta Conmutada de Etiquetas)

El LSP es la ruta construida por uno o más LSRs dentro de un nivel jerárquico por el que un paquete etiquetado y perteneciente a una determinada clase puede circular. En MPLS todos los paquetes pertenecientes a una misma clase equivalente de envío circulan por el mismo LSP.

1.2.2.5 Forwading Equivalence Class (FEC) – (Clase Equivalente de Envío)

La Clase Equivalente de Envío (FEC) hace referencia a un subconjunto de paquetes IP que son tratados de la misma forma por un ruteador (sobre la misma ruta y con el mismo tratamiento de envío). Se puede decir que en el enrutamiento convencional, cada paquete está asociado a un nuevo FEC en cada salto. En MPLS la operación de asignar una FEC a un paquete solo se realiza una vez que el mismo ingrese a la red.

1.2.3 Arquitectura MPLS

Como en toda nueva tecnología, los elementos que definen la arquitectura de la misma deben ser estudiados intensamente ya que cumplen ciertas funciones y roles dentro de un dominio nuevo como es el caso de MPLS.

Elementos como LSRs y LERs son los principales dispositivos que ocupan una parte muy importante al momento de definir la arquitectura MPLS. Estos dispositivos cumplen con la función de intercambiar etiquetas dentro de una red MPLS. Los LSRs y LERs en su estructura interna constan de dos

componentes que requieren de profundo entendimiento como lo son: El plano de Control y el Plano de Datos o de Envío.

1.2.3.1 Plano de Control de conmutación de etiquetas

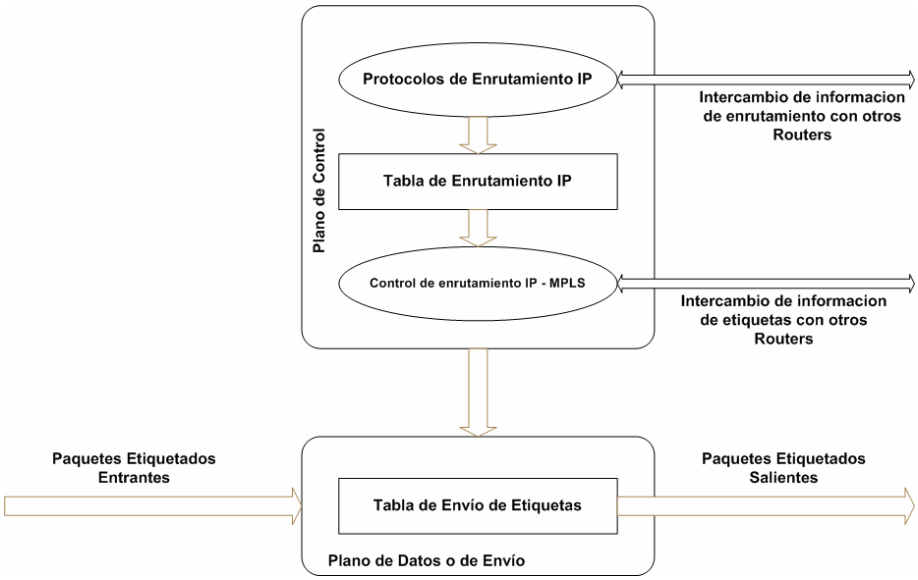


Figura 1-6.- Plano de Control y Plano de Envio

En el Plano de Control en un LSR y un LER se encuentran los protocolos de encaminamiento y las tablas de encaminamiento.

El protocolo de encaminamiento se encarga de mantener la información de las actualizaciones de rutas entre los LSRs que se encuentran dentro de la red MPLS (fig. 1-6). Los protocolos de encaminamiento crean la tabla de enrutamiento IP que es usada para construir la base de información de envío (FIB). Esta tabla de enrutamiento IP en el plano de control es empleada para determinar el intercambio de etiquetas, donde los nodos adyacentes las intercambian para todas las subredes que están contenidas dentro de su tabla. Este intercambio realizado por el protocolo de distribución de etiquetas (LDP) crea la base de información de etiquetas (LIB).

1.2.3.2 Plano de Datos o Plano de Envío de Etiquetas

A diferencia del plano de control, el plano de Envío en los LSRs y LERs difiere un poco ya que en el LER se extienden las funcionalidades debido a que no solo cuenta con la tabla de envío de etiquetas sino que también trabaja con una tabla de envío IP como se puede notar (fig. 1-7)

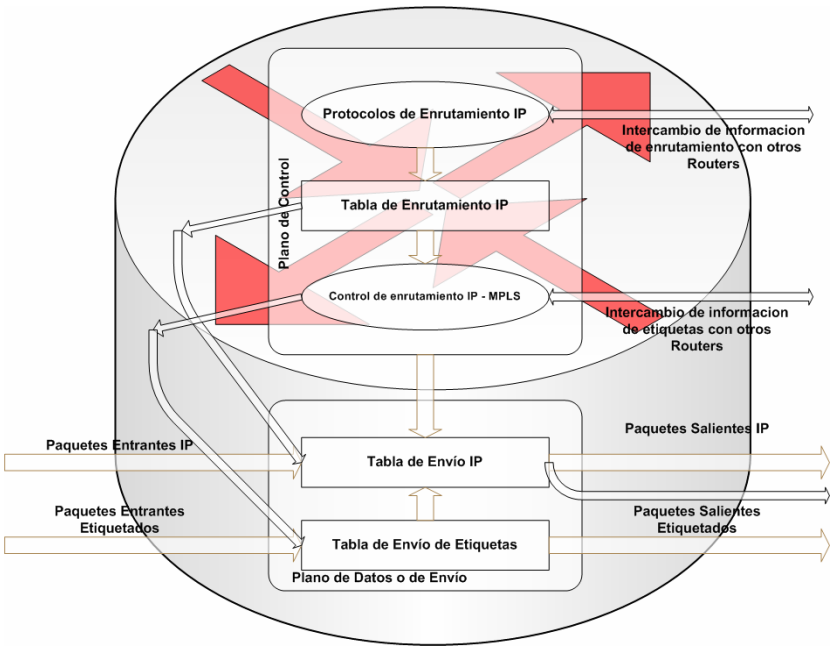


Figura 1-7.- Diagrama de Bloques del LER

Por esta diferencia se describe separadamente las funcionalidades del plano de envío para el LSR y el LER.

LSR: En el proceso de enrutamiento IP - MPLS se utilizan las etiquetas que se intercambian entre LSRs adyacentes que ayudan a la creación de la tabla de envío de etiquetas en el Plano de Datos para enviar los paquetes etiquetados a través de una red MPLS.

LER: La tabla de envío IP estándar es construida en base a tabla de enrutamiento IP y es extendida con información de etiquetas. Esta extensión de componentes en el plano de datos se debe a que los paquetes IP entrantes a un LER pueden ser enviados como paquetes IP natos a un nodo no MPLS o pueden ser etiquetados y enviados a otros nodos MPLS. Además si los paquetes entrantes vienen etiquetados pueden ser enviados a otros nodos MPLS, o si su destino es un dominio no MPLS, su etiqueta puede ser removida y el chequeo de capa de red es realizado (envío IP) para encontrar el destino no MPLS.

En general, y para ambos casos, al crearse la tabla de envío de etiquetas cada entrada de la tabla contendrá una etiqueta de entrada y una etiqueta de salida, que corresponden a cada interfaz de entrada a un nodo MPLS. En el siguiente ejemplo se ilustra el funcionamiento de un LSR del núcleo MPLS. En este caso un paquete que llega a un LSR por la interfaz 3 y con etiqueta 45, se le remueve esa etiqueta y se le asigna la etiqueta 22 que le indica que el paquete debe salir por la interfaz 4 hacia el siguiente LSR, de acuerdo con la información de la tabla.

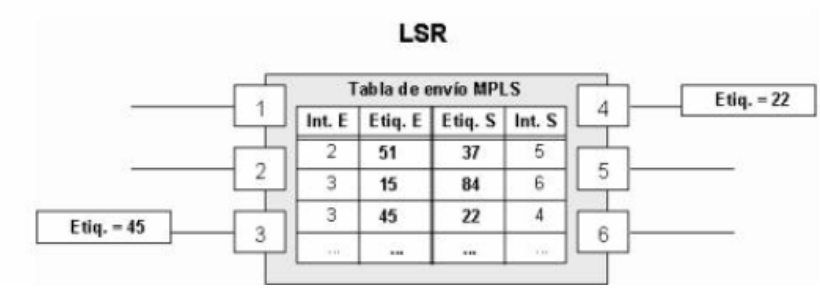


Figura 1-8.- Tabla de Envío MPLS

La clasificación de los paquetes es realizada a la entrada del dominio para así poder realizar la asignación de la etiqueta en la frontera MPLS. Por medio de la grafica (fig. 1-9) se puede notar que un paquete IP al momento de ingresar a una Red MPLS es encapsulado, ya que dentro de un LSR del

núcleo MPLS se ignora la cabecera IP y solamente se toma en cuenta la información de la etiqueta de entrada, se consulta la tabla correspondiente, y la etiqueta es reemplazada por otra nueva de acuerdo con el algoritmo de intercambio de etiquetas.

Al llegar el paquete al LSR de salida o LER de egreso, ve que el siguiente salto lo saca del dominio MPLS; y al consultar la tabla de conmutación de etiquetas se remueve la etiqueta y se envía el paquete por enrutamiento convencional.

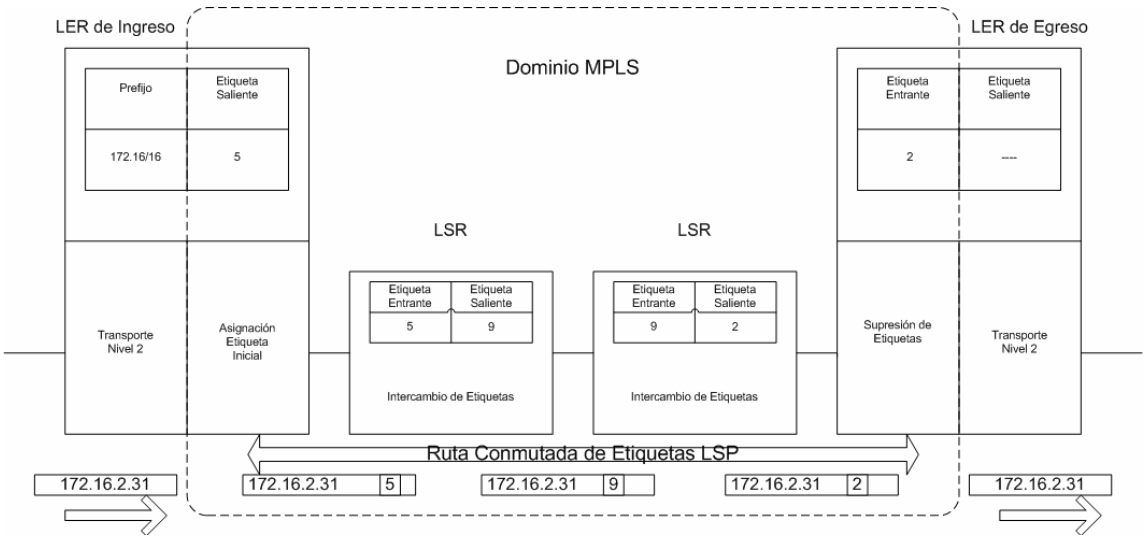


Figura 1-9.- Encapsulamiento y etiquetado de Paquetes

1.2.3.3 Empleo de Etiquetas

El objetivo del empleo de etiquetas en MPLS es realizar la conmutación de las mismas asociándolas a clases equivalentes de envío (FEC), y empleando el valor de ellas para enviar paquetes, incluyendo la determinación del valor de cualquier etiqueta de reemplazo. A continuación

(fig. 1-10) se muestra la relación de la cabecera MPLS con las cabeceras de otros niveles.

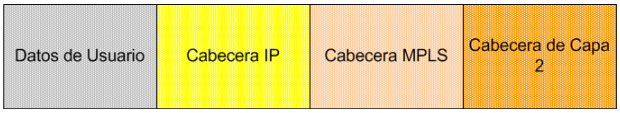


Figura 1-10.- Relación del nivel MPLS con otros niveles

1.2.3.3.1 Header MPLS

La grafica (fig. 1-11) representa el esquema de los campos de la cabecera MPLS.

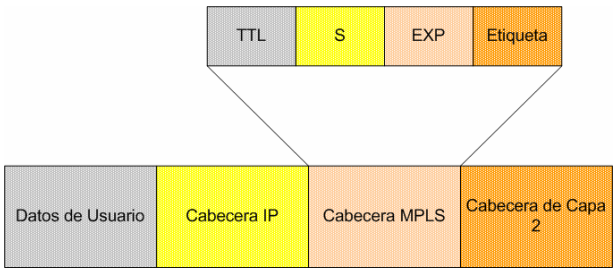


Figura 1-11.- Cabecera MPLS

Según se muestra, la cabecera consta de 32 *bits*, que se distribuyen de la siguiente forma: 20 *bits* para la etiqueta MPLS, 3 *bits* para identificar la clase de servicio en el campo EXP, 1 *bit* de Pila *(Stack)* para poder apilar etiquetas de forma jerárquica en el campo S, y 8 *bits* para identificar el campo TTL (Tiempo de existencia), que sustenta la funcionalidad estándar TTL de las redes IP. De este modo, las cabeceras MPLS permiten cualquier tecnología o combinación de tecnologías de transporte, con la flexibilidad que esto supone para un proveedor IP a la hora de extender su red con servicios de MPLS.

1.2.3.3.2 Pila de Etiquetas

En lo que respecta a MPLS siempre resulta útil tener un modelo más general en el cual, el paquete etiquetado transporte un cierto número de etiquetas, organizadas en una estructura de pila. A esta estructura se le denomina Pila de etiquetas. Aunque MPLS soporta una estructura jerarquizada, el procesamiento de un paquete IP etiquetado es completamente independiente del nivel jerárquico. El procesamiento está siempre basado en la etiqueta del tope de la pila, sin tener en cuenta que cierto número de etiquetas puedan haber estado sobre ella en la pila, anteriormente, o que otras tantas estén bajo ella actualmente.

1.2.3.4 Next Hop Label Forwading Entry (NHLFE) – (Tabla de entrada de envío de etiquetas al siguiente salto)

La tabla de entrada de envío de etiquetas al siguiente salto se emplea cuando un paquete etiquetado es enviado. En ella se puede tener la siguiente información:

- El siguiente salto del paquete.

- La operación de manejar la pila de etiquetas del paquete; la misma que consiste en las operaciones detalladas a continuación:

    - Sustitución de una etiqueta del tope de la pila con una nueva etiqueta específica o con una pila de etiquetas de cierto nivel de profundidad.

    - Extraer de la pila.

Es necesario conocer que la tabla de entrada de envío de etiquetas al siguiente salto contiene además:

- El encapsulado del enlace de datos a emplear cuando se transmite el paquete.

- El método para codificar la pila de etiquetas cuando se transmita el paquete.

- Información extra para manejar el paquete adecuadamente.

Se debe tener en cuenta que en un LSR el siguiente salto podría ser, él mismo. En este caso, el LSR necesita extraer la etiqueta del tope de la pila y se auto enviará el paquete resultante, para lo cual tendría que realizar otra decisión de envío basada en la información que permanezca tras haber extraído la etiqueta de la pila ya que podría darse el caso de tener otro paquete etiquetado o el paquete IP nativo, lo cual conlleva a que, el LSR puede necesitar operar con la cabecera IP para poder reenviar el paquete.

1.2.3.5 Incoming Label Map (ILM) – (Mapa de Etiquetas entrantes)

El mapa de etiquetas entrantes, como su nombre lo indica, mapea cada etiqueta que ingrese a un conjunto de NHLFEs y es muy empleado cuando se reenvían paquetes que llegan etiquetados al LSR.

Dado que el ILM mapea una etiqueta en particular a un conjunto de NHLFEs que contienen mas de un elemento, exactamente uno de los elementos de ese conjunto debe ser elegido antes de que se reenvíe el paquete. Los procedimientos de elección del conjunto no se han definido aún. El mapeado

de una etiqueta hacia un conjunto que contenga más de un NHLFE puede ser útil si se desea balancear la carga de una red sobre múltiples rutas con el mismo costo.

1.2.3.6 Forwading Equivalence Class to Next Hop Label Forwading Entry (FEC to NHLFE) - (FTN)

Este mapa (FTN) asocia cada clase equivalente de envío con un conjunto de NHLFEs y se emplea para reenviar paquetes que lleguen sin etiquetar, pero que deben ser etiquetados antes de ser reenviados. Su comportamiento es análogo al ILM y se puede notar el claro ejemplo cuando ingresan los paquetes sin etiquetar a un *Edge – LSR* o LER de ingreso.

2 DOMINIO MPLS Y DESCRIPCION FUNCIONAL

Como toda arquitectura de Red, el dominio MPLS consta de dispositivos de alta importancia que cumplen sus funciones específicas. Aquellos dispositivos participan en lo que corresponde al encaminamiento de datos.

2.1 Dominio MPLS

El dominio MPLS puede describirse como « Un continuo conjunto de nodos, los cuales operan con enrutamiento y envío MPLS ». Este dominio es típicamente gestionado y controlado por una administración. El concepto de dominio MPLS es algo similar a la noción de un sistema autónomo (AS), tal como es usado este término en el enrutamiento IP convencional, lo cual indica que el sistema autónomo es un conjunto de dispositivos que usualmente se encuentran bajo una administración en común.

El dominio MPLS puede ser dividido en dos partes las cuales son: el núcleo o *core* MPLS y la frontera MPLS.

El núcleo o *core* MPLS consiste de nodos que son vecinos y trabajan con capacidades únicamente MPLS, mientras que la frontera o *edge,* consiste de nodos que son vecinos a dominios MPLS y no MPLS y tienen capacidades tanto de envío MPLS como de envío IP. Los nodos en el núcleo MPLS son llamados ruteadores del proveedor o *P – Routers* y son los comúnmente conocidos LSRs de tránsito.

Los nodos en las fronteras MPLS son conocidos como *Provider – Edge Routers* y son también llamados LERs o Edge – LSRs. Si un LER es el

primer nodo en la ruta para que un paquete viaje a lo largo de un dominio MPLS, este nodo es conocido como LER de ingreso. Si un LER es el último nodo en una ruta, este es conocido como LER de egreso. Estos términos para los LER (ingreso y egreso), son aplicados de acuerdo a la dirección del flujo de datos en un dominio MPLS.

2.1.1 Provider Edge Routers (PEs) – (Routers de frontera al dominio MPLS)

Como se conoce, un *Provider Edge Router* o LER puede estar, ya sea en el ingreso o en el egreso de una nube MPLS. Ellos cuentan con funcionalidades que le permiten trabajar tanto en un dominio MPLS como en un dominio IP. Estos dispositivos permiten el ingreso y salida de la red, y es por ello que entre sus funciones principales está la de recibir paquetes entrantes sin importar que estos vengan etiquetados o sin etiquetar, y darles el tratamiento específico. Cuando un paquete ingresa al dominio MPLS sin etiquetar, el LER de ingreso después de realizar el chequeo correspondiente en su tabla de enrutamiento IP y su tabla de etiquetas, asigna la etiqueta correspondiente a la clase o FEC del paquete y le indica su siguiente LSR que debe atravesar, en otras palabras le indica el LSP o camino que debe seguir.

Cuando un paquete ingresa etiquetado a un LER de salida en la red MPLS, este dispositivo de red le realiza el chequeo correspondiente y al consultar con su tabla de etiquetas el LER de salida decide si el siguiente salto lo saca o no del dominio MPLS. Si el siguiente salto lo saca de la nube, el LER de salida removerá la etiqueta del paquete y lo enviara a su destino por enrutamiento convencional. Si el siguiente salto es una nube MPLS (puede

ser él mismo LER, u otro dominio MPLS) el LER de salida mediante su tabla de etiquetas tomará la decisión correspondiente.

2.1.2 Provider Routers (P – Routers) – (Routers de Core)

Un *Provider – Router* o LSR de tránsito se encuentra en el núcleo de una red MPLS. Son ellos los encargados de recibir la información (paquetes etiquetados MPLS) proveniente de los LER o de otros LSRs y realizar el respectivo encaminamiento mediante la conmutación de etiquetas. A diferencia de los LER, los LSR de tránsito no realizan un chequeo de capa de red, ya que simplemente estudian la etiqueta contenida en la cabecera MPLS dentro de un paquete, y para el envío remueven la etiqueta o etiqueta tope (si el paquete tiene una pila de etiquetas) y reemplazan la misma por otra de acuerdo con la información obtenida de su mapa de etiquetas para indicar el siguiente salto en el LSP, al paquete etiquetado.

2.2 Descripción Funcional de MPLS

La operación de la tecnología MPLS se basa en las componentes funcionales de envío y control, detalladas tempranamente, y que actúan ligadas entre si. Para comprender la forma en que MPLS rompe con los paradigmas de barrido *(swapping)* de etiquetas en capa de enlace y enrutamiento en capa de red, es necesario entender claramente la arquitectura concerniente a los planos de control y de envío ya descritos previamente.

2.2.1 Envío de Paquetes

La base de MPLS está en la asignación e intercambio de etiquetas que ya se conoce, lo cual permite el establecimiento de las rutas por la red. En MPLS las rutas se establecen para un sentido del tráfico, por lo que, para un tráfico en dos sentidos se tiene que establecer dos caminos, uno en cada sentido. Los caminos en MPLS se crean a base de unir uno o más saltos *(hops)* en los que se intercambian las etiquetas mediante protocolos de señalización *(ver sección LDP, RSVP)*, de modo que cada paquete se envía de un LSR a otro, a través del dominio MPLS.

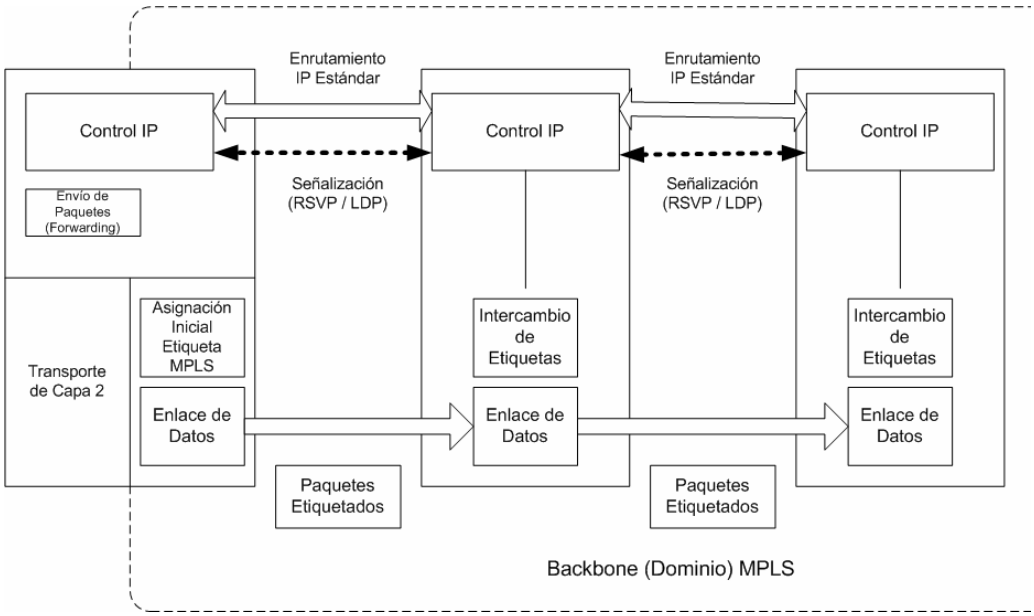


Figura 2-1.- Envío MPLS

Por medio de la grafica (fig. 2-1) se ilustra la funcionalidad de la tecnología MPLS. El envío se emplea mediante el intercambio de etiquetas en las rutas, para ello, MPLS utiliza protocolos de señalización como RSVP y LDP.

2.2.2 Control de la información

En primera instancia, en lo que se refiere al control de la información, MPLS requiere de algoritmos de control para establecer los caminos conmutados de etiquetas LSPs. Se utiliza la información de encaminamiento que manejan los protocolos internos (IGPs) para construir las tablas de encaminamiento, lo cual, MPLS aprovecha, y para cada ruta IP en la red crea un LSP a base de concatenar las entradas y salidas en cada tabla de los LSRs. El protocolo interno se encarga de pasar la información necesaria.

En segunda instancia se trata de mantener lo que es la señalización, que es utilizada para marcar el camino siempre que se requiera establecer un circuito virtual para la distribución de etiquetas entre los LSRs. Para la señalización MPLS utiliza los protocolos RSVP del modelo de servicios integrados del IETF y el protocolo LDP

Una vez presentados y comprendidos los componentes funcionales de MPLS, el esquema de funcionamiento mostrado a continuación (fig. 2-2) deja reflejadas las diversas funciones en cada uno de los elementos que integran un dominio MPLS. Es importante destacar que en la frontera de una nube MPLS se tiene una red convencional de ruteadores IP. El núcleo proporciona una arquitectura de transporte que hace aparecer a un par de ruteadores a una distancia de un solo salto, lo cual indica que funcionalmente estos dispositivos están unidos todos en una topología completamente mallada. La unión a un solo salto se realiza por MPLS mediante los correspondientes LSPs.

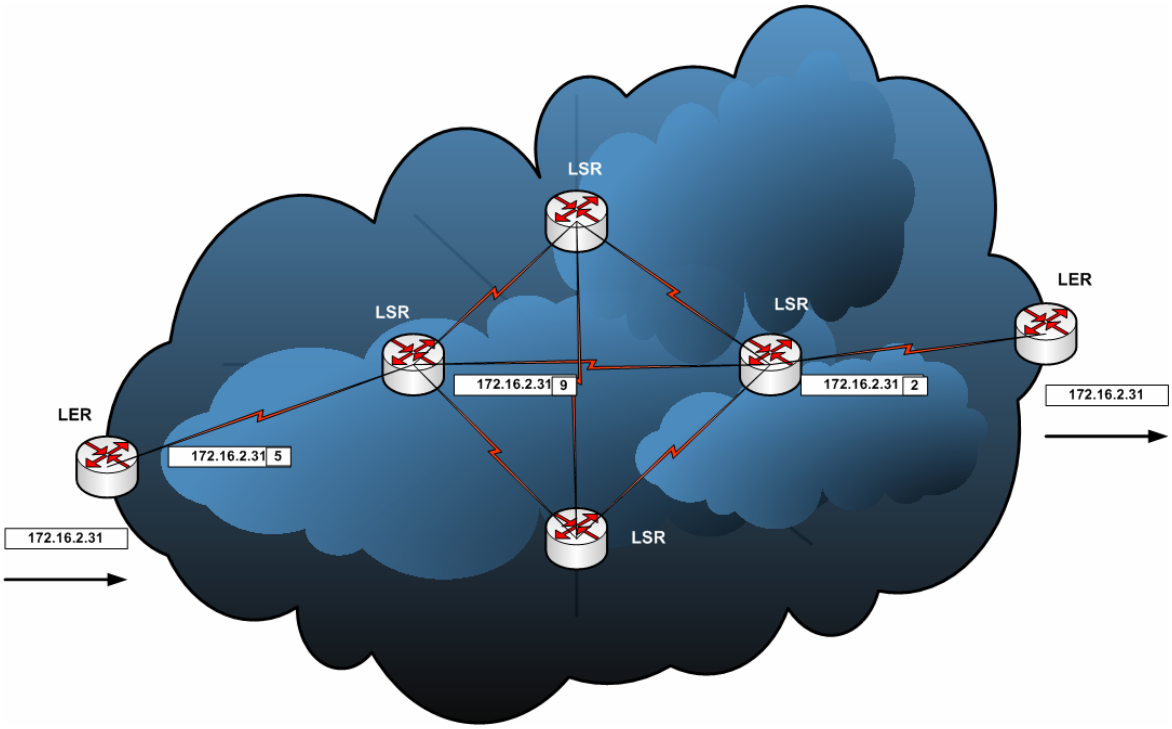


Figura 2-2.- Viaje de un paquete por un dominio MPLS

2.3 Métodos de solicitud de Etiquetas

Para entrar en detalles correspondientes a los métodos de solicitud de etiquetas es necesario conocer y entender el término *"Downstream".* Este término es de importancia ya que indica la dirección en la que viaja el flujo de datos. La dirección del flujo de datos en conocida como *"Dirección Downstream"*. Se define como vecino *Downstream* a un LSR el cual recibe el flujo del tráfico. Es decir si tenemos un ruteador A enviando información a un ruteador B, entonces B es vecino *Downstream* de A, y A es por ende vecino *Upstream* de B.

Al ingresar un paquete IP a un dominio MPLS, el LER de ingreso en la red negocia la etiqueta con sus LSRs de *Downstream* y envía el paquete al LSR

correspondiente el cual también negocia la etiqueta con sus vecinos *Downstream* correspondientes. Entre los métodos de negociación de las etiquetas se conocen principalmente el *Downstream bajo demanda* y el *Downstream sin solicitar.*

2.3.1 Downstream bajo Demanda

La arquitectura MPLS permite a un LSR pedir de forma explícita, a su siguiente punto, una etiqueta específica para una determinada clase equivalente de envío (FEC), lo que quiere decir que, los LSRs *Upstream* solicitan etiquetas a sus vecinos *Downstream* y estos distribuyen las etiquetas sobre la demanda. En la gráfica (fig. 2-3) puede observarse un claro ejemplo de este método de solicitud de etiquetas.

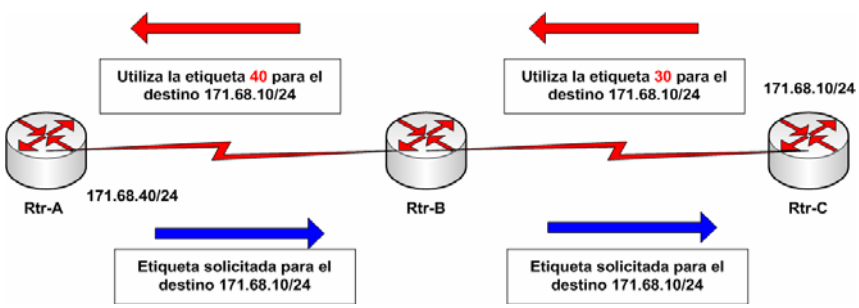


Figura 2-3.- Técnica de Downstream bajo demanda

2.3.2 Downstream sin Solicitar

En MPLS también está permitido distribuir los enlaces ya efectuados FEC/etiqueta a LSRs que no han solicitado de forma explícita la etiqueta. En este método los LSRs distribuyen sus etiquetas a cada vecino *Upstream* sin que éste la haya solicitado previamente. Con la figura (fig. 2-4) puede observarse un claro ejemplo de la distribución de etiquetar sin solicitar.

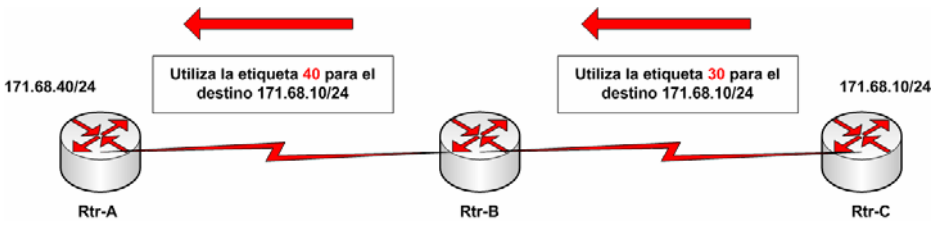


Figura 2-4.- Técnica de Downstream sin solicitar

Cabe mencionar que algunas implementaciones de MPLS pueden optar por llevar ambos procedimientos implementados, pero aquello vendrá dado por las características de las interfaces, ya que al convivir las dos técnicas en una red se debe asegurar que los puntos adyacentes siempre se pongan de acuerdo con el método que trabajarán.

En MPLS la decisión de enlazar cierta etiqueta a un FEC la realiza el LSR que es *Downstream* con respecto a un enlace específico. Por esta razón se entiende que la distribución de etiquetas se realiza desde la dirección *Downstream* hacia *Upstream.*

2.4 Label Switched Path (LSP) - (Ruta Conmutada de Etiquetas)

El camino o ruta conmutada de etiquetas, es el circuito virtual que siguen en una red MPLS todos los paquetes asignados a un mismo FEC. Los LSPs *simplex* (en una dirección) por naturaleza, se crean gracias a la unión de LSRs que intercambian etiquetas. Al primer LSR que interviene en un LSP se le denomina LSR de entrada o de cabecera y al último se le denomina de salida o de cola. Los dos se encuentran en la frontera del dominio MPLS como ya se mencionó anteriormente, mientras que el resto de LSRs que

forman el LSP son LSRs internos del dominio MPLS. Un LSR funciona como un ruteador pero a base de intercambio de etiquetas según la tabla de envío que se construye a partir de la información que proporciona la componente de control. Cada tabla de cada LSR en la red contiene etiquetas tanto de entrada como de salida de cada nodo, de manera que mediante el intercambio de estas etiquetas se logra construir el camino de LSRs por el cual viajan los paquetes etiquetados (LSP).

2.4.1 Protocolos de distribución de etiquetas MPLS

Dado que los caminos conmutados de etiquetas (LSP) son construidos mediante la concatenación de los LSRs. Estos LSRs construyen los LSPs mediante el intercambio de etiquetas que es realizado por los protocolos de señalización tales como: LDP y RSVP.

2.4.1.1 Label Distribution Protocol (LDP)

El protocolo LDP es uno de los protocolos de distribución de etiquetas que ha sido creado con este propósito. Es un conjunto de procedimientos mediante los cuales los LSRs crean los caminos de conmutación de etiquetas (LSPs) a través de una red MPLS. Estos caminos pueden acabar en vecinos conectados entre sí directamente (como IP salto a salto), o pueden acabar en un nodo de salida de cierta red, activando así la conmutación entre todos los nodos intermedios de un dominio.

Los LDPs asocian una FEC con cada camino LSP que sea creado, dando a entender que los FECs asociados con algún LSP, especifican los paquetes IP que viajarán por ese camino.

Es importante conocer el trato que se les da a dos dispositivos que utilizan el protocolo LDP para establecer una sesión, y; es por esta razón que dos LSRs que estén utilizando el protocolo LDP para intercambiar información de asociación de etiqueta/FEC, son conocidos como pares LDP con respecto a esa información. Entre ellos se mantiene una sesión LDP que permite a cada par aprender la información de las etiquetas del otro. El protocolo LDP es bidireccional.

Entre los pares LDP existe un intercambio de mensajes LDP los cuales se pueden clasificar en cuatro categorías:

a) **Mensajes de Descubrimiento:** Empleados para mantener y anunciar la presencia de un LSR en la red.

b) **Mensajes de Sesión:** Empleados para establecer, mantener y analizar las sesiones entre los pares LDP.

c) **Mensajes de Anuncio:** Empleados para crear, cambiar y borrar asociaciones de etiquetas con FECs

d) **Mensajes de Notificación:** Empleados para dar información de aviso o de error.

Los mensajes de descubrimiento anuncian la presencia de un LSR en la red, lo cual se realiza enviando un mensaje *HELLO* periódicamente. Cuando un LSR desea establecer una sesión con otro LSR, aprendido gracias al *HELLO*, empleará el proceso de inicialización LDP sobre TCP. Si se lleva a cabo de forma correcta el procedimiento de inicialización LDP, los dos LSRs serán pares LDP y podrán intercambiar información de anuncio.

El funcionamiento correcto del protocolo LDP requiere una recepción fiable y ordenada de mensajes. Para ello, se emplea el protocolo TCP para mensajes de sesión, de anuncio, y notificación, excepto para mensajes de descubrimiento que viajan sobre UDP.

En cuanto a la estructuración de un mensaje LDP se conoce que su estructura emplea una metodología de codificación TLV *(Type – Lenght - Value)* – (Tipo – Longitud – Valor). Las sesiones LDP no solo se dan entre LSRs conectados directamente, sino también entre LSRs que no están conectados directamente, y para llevar a cabo estas sesiones LDP, deben darse ciertas características que se detallan mediante el siguiente ejemplo:

Considere una aplicación en la que un LSRa envía el tráfico a un LSRb que no está directamente conectado a él. El camino entre LSRa y LSRb incluirá varios LSRs intermedios (LSR_1…LSRn). Se crearía una sesión entre LSRa y LSRb que permitirá al LSRb conmutar el tráfico etiquetado procedente del camino con LSRa permitiendo a LSRb emplear métodos para anunciar a LSRa las etiquetas para este propósito.

En esta situación LSRa aplicará una pila de etiquetas de nivel 2 al paquete para enviarlo al LSRb:

- La primera etiqueta aprendida desde LSR_1, le permitirá enviar el tráfico por el camino de LSRs que existe entre LSRa y LSRb.

- Una segunda etiqueta aprendida de LSRb para permitir al LSRb conmutar el tráfico etiquetado que llegue por el LSP.

Para resumir el procedimiento, LSRa primero añade la etiqueta aprendida en su sesión con LSRb a la pila de etiquetas del paquete, finalmente añade a la

pila una segunda etiqueta aprendida de LSR_1 para entrar en el camino que forman los LSRs entre LSRa y LSRb.

2.4.1.1.1 Descubrimiento LDP

Se conoce como descubrimiento al mecanismo mediante el cual un LSR conoce a otro LSR y entre ellos se forman pares LDP. Debido al descubrimiento no es necesario configurar explícitamente los pares LDP.

De acuerdo a la forma que se requiera establecer los pares LDP, se conocen dos mecanismos de descubrimiento, el descubrimiento básico y el descubrimiento extendido.

- **Descubrimiento Básico:** En este mecanismo de descubrimiento un LSR envía periódicamente HELLOs de enlace como paquetes UDP al puerto LDP de descubrimiento, con la dirección multicast del grupo de todos los routers en una subred. El mensaje HELLO lleva el identificador LDP para el espacio de etiquetas que el LSR trata de utilizar, y otra posible información adicional. La recepción del mensaje HELLO identifica una adyacencia con algún posible par LDP accesible a nivel de enlace así como el posible espacio de etiquetas que el par LDP trata de emplear.

- **Descubrimiento Extendido:** Este mecanismo se emplea para sesiones LDP entre LSRs que no están conectados directamente, para lo cual un LSR envía HELLOs direccionados a una dirección IP específica como paquetes UDP hacia el puerto de aquella dirección IP. El HELLO lleva el identificador para el espacio de etiquetas que el LSR trata de utilizar. Como en el caso del descubrimiento básico, la recepción del mensaje HELLO identifica una adyacencia con algún par LDP accesible a nivel de

enlace, así como el espacio de etiquetas que el par LDP trata de emplear.

2.4.1.1.2 Establecimiento y Mantenimiento de Sesiones LDP

Una vez analizado el descubrimiento, se procede al establecimiento de una sesión entre pares LDP, en donde se deben considerar dos aspectos sumamente importantes como lo son:

a) El establecimiento de la conexión de transporte.

b) La inicialización de la sesión.

- **El establecimiento de la conexión de transporte**: Para el establecimiento de una sesión LDP entre dos LSR (suponer LSR_1 y LSR_2) y para espacios de etiquetas “a” y “b” respectivamente, LSR_1 intentará abrir una conexión TCP para tener una sesión LDP con LSR_2 de la siguiente manera:

    1) LSR_1 determina las direcciones de transporte a emplear: A_1 (por él mismo) y A_2 por el LSR_2.

    2) La dirección A_1 se determina empleando TLV en los HELLOs que envía a LSR_2 para anunciar una dirección, en este caso A_1.

    3) De la misma forma A_2 se determinará de manera análoga empleando los HELLOs de LSR_2.

LSR_1 determina si su papel será activo o pasivo en el establecimiento de la sesión comparando los valores de A_1 y A_2. Si $A_1>A_2$, entonces LSR_1 tendrá un rol activo en la conexión, caso contrario será pasivo.

Si LSR_1 es activo, intentará establecer la conexión TCP/LDP conectándose con el puerto LDP configurado en la dirección A_2. Si LSR_1 es pasivo, entonces esperará que LSR_2 establezca la conexión TCP con su puerto LDP.

- **Inicialización de Sesión:** Luego de establecerse la conexión TCP, los pares negocian los parámetros de inicialización, intercambiando mensajes de inicialización LDP. Estos parámetros incluyen: versión del protocolo LDP, método de distribución de etiquetas, valores de temporizadores.

  Si la negociación de los parámetros de inicialización ha tenido éxito, se establece la sesión entre los dos pares como se indica a continuación:

  Si LSR_1 juega un rol activo, inicia la negociación de los parámetros de sesión enviando un mensaje de inicialización a LSR_2, caso contrario esperará que LSR_2 inicie la negociación.

Para el mantenimiento de las sesiones LDP, éste protocolo emplea la recepción regular de PDUs LDP en la conexión de transporte de la sesión y lo hace con el fin de monitorear la integridad de dicha sesión. Un LSR mantiene un temporizador *KeepAlive* (sigo con vida) para cada par, si este temporizador llegara a expirar sin haber recibido una PDU LDP del par, entonces el LSR concluye que la conexión TCP está incorrecta, o que el par ha fallado y termina la sesión cerrando la conexión TCP.

2.4.1.2 Resource Reservation Protocol (RSVP)

Otro de los protocolos que se utiliza con la finalidad de brindar señalización es el Protocolo de reserva de recursos RSVP que pertenece al modelo de servicios integrados del IETF.

RSVP utiliza mecanismos para establecer LSPs, distribuir etiquetas y realizar muchos trabajos relacionados con el uso de las mismas para satisfacer los requerimientos que demanda la ingeniería de tráfico.

Es un protocolo de estado de enlace, lo cual significa que cuando una ruta está siendo configurada, ella tiene que ser continuamente actualizada para mantener los recursos reservados. Se conoce que es además un protocolo receptor – orientado, lo cual da a entender que las peticiones para la reservación son hechas desde el receptor final de una ruta.

Cuando RSVP es utilizado para la configuración de un LSP, el ruteador de ingreso empieza enviando un mensaje *PATH* en el camino donde un LSP será configurado. Cada LSR de tránsito en dicha ruta tiene que chequear si existe la posibilidad de configurar el LSP pedido. En caso que el LSP sea rechazado, un mensaje de error es retornado en dirección *Upstream* hasta que dicho mensaje alcance el ruteador de ingreso. De cualquier manera, el mensaje PATH es enviado al siguiente LSR de tránsito en la ruta hasta que éste alcance el ruteador de egreso. Luego que el ruteador de egreso recibe el mensaje PATH, éste responde con un mensaje *RESV* en la dirección *Upstream* a través de la cual el mensaje *PATH* viajó. En el mensaje *RESV*, los LSRs de *Downstream* incluyen la etiqueta que ellos quieren que el LSR de *Upstream* adyacente utilice para el LSP que está siendo configurado.

Ninguna reservación es realizada en los LSRs hasta que el mensaje RESV es retornado.

2.5 Protocolos de Enrutamiento Dinámico para MPLS

MPLS al igual que las tecnologías actuales de TCP/IP, utiliza los protocolos de enrutamiento dinámico tales como: protocolos de Gateway Interior y Exterior IGPs y EGPs respectivamente.

El enfoque principal en este trabajo está en los protocolos de enrutamiento dinámico que se utilizan en el diseño de la red MPLS que se describirá en secciones posteriores, para lo cual se diseña la topología usando principalmente OSPF y BGP.

2.5.1 Border Gateway Protocol (BGP)

Para comprender correctamente la labor del protocolo de enrutamiento de Gateway de Frontera, se debe tener claro el concepto de lo que significa un sistema autónomo en las redes de datos.

Un sistema autónomo no es más que un conjunto de dispositivos de *networking* (Ruteadores, Conmutadores, etc.) que se encuentran bajo una administración común. Teniendo claro el concepto de los sistemas autónomos se puede llegar al correcto entendimiento de las labores que realiza el protocolo BGP.

El protocolo BGP es un protocolo de enrutamiento utilizado para el intercambio de información entre sistemas autónomos, y el claro ejemplo de

ello se puede notar en el intercambio de información de la Internet, lo cual se hace entre los ISPs.

Cuando BGP es usado entre los sistemas autónomos, entonces el protocolo es referido como BGP externo (E – BGP). Si un proveedor de servicio está usando BGP para intercambiar rutas dentro de un sistema autónomo, entonces el protocolo es llamado BGP interior (IBGP). Al ser utilizado para el intercambio de rutas dentro de un sistema autónomo surge la aplicación de redes privadas virtuales las cuales son configuradas en las fronteras de un sistema autónomo. El gráfico (fig. 2-5) muestra brevemente la definición de sistemas autónomos.

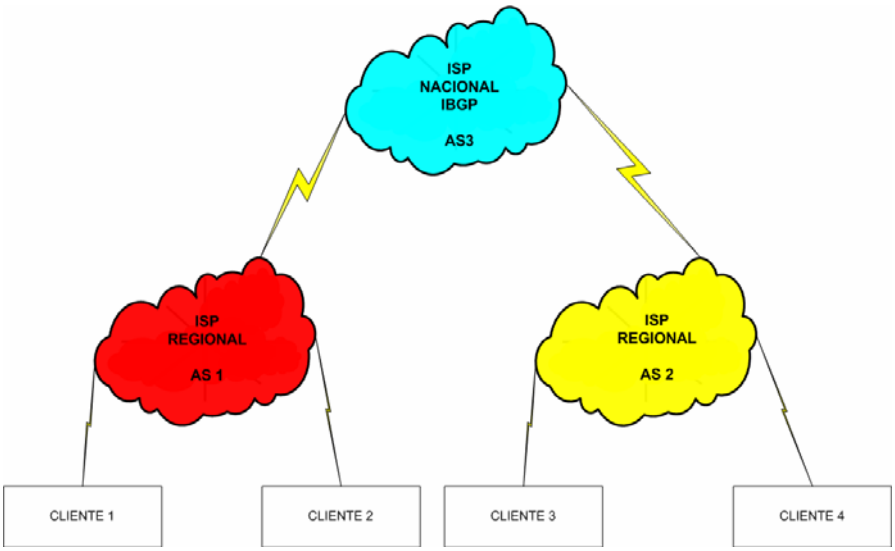


Figura 2-5.- Sistemas Autónomos

En la actualidad la distribución más usual del protocolo de enrutamiento BGP es la versión 4 con extensiones multiprotocolo, mas conocida como *Multi – Protocol* BGP (MP – BGP), la cual se describe a continuación.

2.5.1.1 MP – BGP (BGPv4)

Anteriormente BGPv4 era capaz de llevar solamente información para tráfico IPv4. Sin embargo como se define en el RFC 2283, ya existen extensiones que permiten que BGPv4 lleve información de enrutamiento para múltiples protocolos de capa de red (ej. IPv6, IPX, etc…). Las extensiones son compatibles con versiones anteriores, es decir, un ruteador que soporte las extensiones puede operar con otro ruteador que no soporte las extensiones.

Las únicas tres partes de información llevadas por BGPv4 que están especificadas en IPv4 son:

a) El atributo de próximo salto *(Next_Hop)*, el cual es expresado como una dirección IPv4.

b) Agregador *(Aggregator)*, que contiene una dirección IPv4.

c) La información alcanzable de capa de red (NLRI) que está expresada como prefijos de direcciones IPv4.

Se asume que algún ruteador que hable BGP debe tener una dirección IPv4, la cual será utilizada entre otras cosas en el atributo Agregador. Además, para lograr que BGPv4 soporte enrutamiento para múltiples protocolos de capa de red se deben añadir únicamente dos cosas:

a) La habilidad para asociar un protocolo de capa de red particular con la información del próximo salto.

b) La habilidad para emparejar un protocolo particular de capa de red con el NLRI. Para identificar los protocolos de capa de red, en el RFC 2283 se utiliza la familia de direcciones tal como se define en el RFC 1700.

Alguien podría además observar que la información del próximo salto (información proporcionada por el atributo Next_Hop) es significativamente necesaria únicamente en conjunto con los avisos de destinos alcanzables, lo que sugiere que estos avisos deberían ser agrupados con los del siguiente salto a ser usado por estos destinos.

Para proporcionar compatibilidad retrospectiva, también como para simplificar la introducción de las capacidades multiprotocolo en BGPv4, en el RFC 2283 se utilizan dos nuevos atributos:

a) Multiprotocolo alcanzable NLRI *(MP_REACH_NLRI)*

b) Multiprotocolo inalcanzable NLRI *(MP_UNREACH_NLRI)*

El primero *(MP_REACH_NLRI)* es utilizado para llevar el conjunto de destinos alcanzables junto con la información del siguiente salto a ser empleada para el envío a estos destinos. El segundo *(MP_UNREACH_NLRI)* se emplea para llevar el conjunto de destinos inalcanzables. Ambos atributos son opcionales y no transitivos. De esta forma un ruteador que utiliza BGP pero que no soporta las capacidades multiprotocolo ignorará la información llevada en estos atributos, y no se pasara esta información a otros ruteadores BGP

Multiprotocolo alcanzable NLRI (MP_REACH_NLRI)

Este es un atributo opcional, no transitivo, que puede ser empleado para los siguientes propósitos:

a) Advertir una ruta factible a un par.

b) Para permitir que un ruteador advierta la dirección de capa de red (IP) de aquel ruteador que debería ser usado como el siguiente salto a los destinos listados en el campo NLRI del atributo MP_NLRI.

c) Para permitir que un ruteador dado reporte algunos o todos los puntos añadidos de subredes *(SNPAs)* que existen dentro del sistema local.

El atributo contiene uno o más triplos <Información de Familia de Direcciones, Información de Próximo Salto, NLRI>, donde cada triplo es codificado como se muestra a continuación:

Identificador de Familia de Direcciones (2 octetos)
Identificador Subsecuente de familia de direcciones (1 octeto)
Longitud de la dirección de red del siguiente salto (1 Octeto)
Dirección de red del siguiente salto (variable)
Numero de SNPAs (1 octeto)
Longitud del primer SNPA (1 octeto)
Primer SNPA (variable)
Longitud del segundo SNPA (1 octeto)
Segundo SNPA (variable)
…
Longitud del último SNPA (1 octeto)
Ultimo SNPA (variable)
NLRI

Tabla 2-1.- MP_REACH_NLRI

El uso y significado de estos campos se detalla a continuación:

Identificador de Familia de Direcciones

Este campo lleva la identidad del protocolo de capa de red asociado con la dirección de red que sigue. Presentemente los valores definidos para este campo se especifican en el RFC 1700 (ver la sección de números de familia de direcciones).

Identificador Subsecuente de Familia de Direcciones

Este campo proporciona información adicional acerca del tipo de NLRI llevada en el atributo.

Longitud de la Dirección de Red del Siguiente Salto

Un campo de 1 octeto cuyo valor expresa la longitud del campo de “dirección de red del siguiente salto” medida en octetos.

Dirección de Red del Siguiente Salto

Es un campo de longitud variable que contiene la dirección del siguiente ruteador en la ruta al sistema destino.

Longitud del n – ésimo SNPA

Campo de 1 octeto cuyo valor expresa la longitud del “n – ésimo SNPA del siguiente salto”

N –ésimo SNPA del siguiente salto

Es un campo de longitud variable que contiene un SNPA del ruteador cuya dirección de red esta contenida en la “Dirección de red del siguiente salto”. La longitud del campo es un número integrado de octetos en lo que a longitud se refiere, conocidamente es el entero redondeado de una mitad de la longitud SNPA expresada en semi – octetos, un valor en este campo será relleno con un semi – octeto rastreador cuyos campos tienen cero.

Información de Alcance de Capa de Red (NLRI)

Este campo de longitud variable lista los NLRI para las rutas factibles, las cuales están siendo advertidas en este atributo. Cuando el campo Identificador Subsecuente de Familia de Direcciones es puesto a uno de los

valores definidos anteriormente, cada NLRI es codificado como se especifica en la sección de “Codificación de NLRI” detallada mas adelante.

La información del siguiente salto llevada en el atributo de ruta MP_REACH_NLRI, define la dirección de capa de red del ruteador de frontera que debería ser empleado como el siguiente salto a los destinos listados en el atributo MP_NLRI en el mensaje de actualización *(UPDATE)*. Cuando se advierte un atributo MP_REACH_NLRI a un par externo, un ruteador puede usar una de sus propias direcciones de interface en el componente siguiente salto del atributo, proporcionando al par externo al cual el ruteador está siendo advertido que comparta una subred común con la dirección del siguiente salto.

Normalmente la dirección del siguiente salto es escogida de tal manera que la ruta mas corta disponible sea tomada. Un ruteador nunca debe advertir la dirección de un par, a otro par al que tenga como siguiente salto, para una ruta que el ruteador BGP está originando. Un ruteador BGP nunca debe instalar una ruta con él mismo como siguiente salto.

Cuando un ruteador BGP advierte la ruta a un par interno, el aviso del ruteador no debería modificar la información del siguiente salto asociada con la ruta. Cuando un ruteador BGP recibe la ruta mediante un enlace interno, este podrá enviar los paquetes a la dirección del siguiente salto si la dirección contenida en el atributo es una subred común con los Routers BGP local y remoto.

Un mensaje de actualización *UPDATE* el cual lleva el MP_REACH_NLRI debe también llevar los atributos *ORIGIN* y *AS_PATH* (tanto en intercambios EBGP como en intercambios IBGP). Además, en los intercambios IBGP tal mensaje debe también llevar el atributo *LOCAL_PREF*. Si tal mensaje es

recibido desde un par externo, el sistema local chequeará si el extremo del sistema autónomo (AS) es igual al número del sistema autónomo del par que envió el mensaje. Si aquello no es el caso, el sistema local enviará el mensaje de notificación *(NOTIFICATION)* con mensajes de error y el código de error debido a la ruta mal formada.

Multiprotocolo inalcanzable NLRI

Este atributo opcional y no transitivo puede ser utilizado para los propósitos de retirar múltiples rutas no factibles del servicio. El atributo contiene uno o más triplos <Información de Familia de Direcciones, Longitud de Rutas no Factibles, Rutas Separadas>, donde cada triplo es codificado como se muestra en la tabla siguiente:

Identificador de Familia de Direcciones (2 octetos)
Identificador Subsecuente de Familia de Direcciones (1 octeto)
Rutas Separadas (variable)

Tabla 2-2.- MP_UNREACH_NLRI

El uso y significado de estos campos se detalla como sigue:

Identificador de Familia de Direcciones

Este campo lleva la identidad del protocolo de capa de red asociado con el NLRI que sigue. Los valores definidos para este campo se los puede encontrar en el RFC 1700 (sección números de familia de direcciones).

Identificador Subsecuente de Familia de Direcciones

Proporciona información adicional acerca del tipo de NLRI llevada en el atributo

Rutas Separadas

Este campo de longitud variable lista los NLRI para las rutas que están siendo separadas del servicio. Cuando un campo identificador subsecuente de familia de direcciones es puesto a uno de los valores definidos en el RFC 2283, cada NLRI es codificado como se especifica en la sección de “Codificación NLRI” detallada mas adelante.

Un mensaje de actualización que contiene el MP_UNREACH_NLRI no es requerido para llevar algún otro atributo de ruta.

Codificacion NLRI

La información alcanzable del siguiente salto (NLRI) es codificada como una o más duplas de la forma <Longitud, Prefijo>, cuyos campos se describen de inmediato:

Longitud (1 octeto)
Prefijo (variable)

Tabla 2-3.- Codificacion NLRI

El uso y significado de los campos se detalla a continuación:

Longitud

Este campo indica la longitud en bits del prefijo dirección. Una longitud de cero indica un prefijo que empareja todas las direcciones.

Prefijo

El campo Prefijo contiene los prefijos de direcciones seguidos por grandes bits de rastreo para hacer que el final del campo caiga en un octeto fronterizo.

Identificador Subsecuente de Familia de Direcciones

En el RFC 2283 se define los siguientes valores para el campo identificador de familia de direcciones llevado en los atributos MP_REACH_NLRI y MP_UNREACH_NLRI:

a) NLRI empleado para envío unicast.

b) NLRI usado para envío multicast.

c) NLRI utilizado tanto para envío unicast y envío multicast.

Es necesario recalcar que la extensión a BGP no cambia las publicaciones de seguridad subyacentes.

2.5.2 Only Shortest Path First (OSPF)

El protocolo de “Solamente la ruta mas corta primero” (OSPF), es un protocolo de estado de enlace. Se conoce que los protocolos de estado de enlace mantienen una base de datos de información de topología. El algoritmo de enrutamiento de estado de enlace mantiene información compleja sobre ruteadores lejanos y su interconexión. Los protocolos de estado de enlace generan una inundación *(flooding)* de información de ruta, que da a cada ruteador una visión completa de la topología de red. El método de actualización desencadenada por eventos permite el uso eficiente de un ancho de banda y una convergencia rápida. Los cambios en el estado de un enlace se envían a todos los ruteadores en la red tan pronto como se produce.

El protocolo OSPF es uno de los protocolos de estado de enlace más importantes, y se basa en las normas de código abierto *(Open Source)*, lo que significa que muchos fabricantes lo pueden desarrollar y mejorar. Es un protocolo complejo que se describe en varios estándares del IETF cuya implementación en redes más amplias representa un verdadero desafió. Este es un protocolo de enrutamiento de gateway interior (IGP) que es preferido por todos ya que presenta soluciones de escalabilidad. OSPF puede ser usado tanto en redes pequeñas como en redes grandes, en una sola área o en varias como puede verse en la gráfica (fig. 2-6).

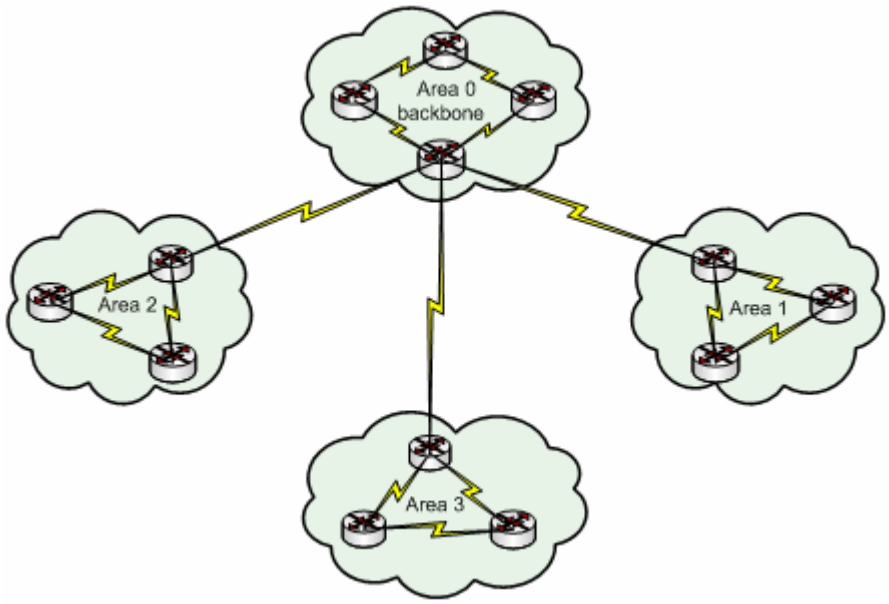


Figura 2-6.- Áreas OSPF

Las grandes redes OSPF utilizan un diseño jerárquico, dado que varias áreas se conectan a un área de distribución o a un área cero, conocida como backbone. El enfoque del diseño para redes OSPF permite el control extenso de las actualizaciones de enrutamiento. La definición de área acelera la convergencia, limita la inestabilidad de la red y mejora el rendimiento.

OSPF utiliza un algoritmo de ruta mas corta desarrollado por Dijkstra, un especialista holandés en informática en 1959. Este algoritmo considera la red como un conjunto de nodos conectados con enlaces punto a punto. Cada enlace tiene un costo, un nombre y cuenta además con una base compleja de todos los enlaces y por lo tanto se conoce la información sobre la topología física en su totalidad. Todas las bases de datos del estado de enlace, dentro de una determinada área, son idénticas. El algoritmo de ruta más corta calcula entonces la topología sin bucles con el nodo como punto

de partida y examinando a su vez la información que posee sobre nodos adyacentes.

Para que los ruteadores OSPF puedan compartir la información de enrutamiento se requiere una relación de vecinos y se tiende a esto cuando un ruteador es adyacente con por lo menos uno en cada red IP a la cual esta conectado. Los ruteadores OSPF determinan con que otros pueden intentar formar adyacencias tomando como base el tipo de red a las cuales están conectados, es decir, unos trataran de hacerse adyacentes con respecto a todos los ruteadores vecinos y otros trataran de hacerse adyacentes con respecto a solo uno de los ruteadores vecinos. Una vez formada la adyacencia, se intercambia la información del estado de enlace. Los equipos de enrutamiento con interfaces OSPF reconocen tres tipos de redes:

a) Multiacceso de *Broadcast* (ej. Ethernet)

b) Redes Punto a Punto

c) Multiacceso sin *Broadcast* (NBMA) – (ej. Frame Relay)

Cuando un ruteador inicia un proceso de enrutamiento OSPF en una interfaz, envía paquetes de descubrimiento *(HELLOs)* a intervalos regulares. En la capa de red, los paquetes de descubrimiento se direccionan hacia la dirección *Multicast* 224.0.0.5 que equivale a todos los ruteadores OSPF, los mismos que utilizan estos paquetes para iniciar nuevas adyacencias y asegurarse que entre los vecinos se mantenga el funcionamiento. Los mensajes de descubrimiento *(HELLOs)* se envían cada 10 segundos por defecto en las redes multiacceso de *broadcast* y punto a punto. En las interfaces que se conectan a las redes NBMA como *Frame Relay*, el tiempo por defecto es de 30 segundos.

En las redes multiacceso el protocolo Hello elige un ruteador designado (DR) el cual se hace adyacente a todos los ruteadores del segmento *broadcast* y presenta un único punto de falla ya que todos los ruteadores del segmento envían el estado de enlace a este ruteador designado. Además de ello se elige un ruteador designado de respaldo (BDR). Aunque el paquete de descubrimiento es pequeño, consiste en un encabezado de paquete OSPF, en donde para el paquete de descubrimiento, el campo de tipo se establece en 1 (fig. 2-7).

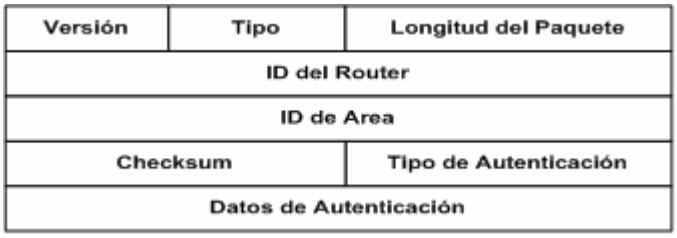


Figura 2-7.- Encabezado de Paquete OSPF

El paquete de descubrimiento transmite información para la cual, todos los vecinos deben estar de acuerdo antes de que se forme una adyacencia y que se pueda intercambiar información del estado de enlace.

Mascara de Red		
Intervalo de Hello	Opciones	Prioridad Del Router
Intervalo Muerto		
Router Designado		
Router Designado de Respaldo		
ID del Router del Vecino		
ID del Router del Vecino		
Se pueden agregar adicionales del ID del Router Vecino al final del Encabezado, de ser necesario		

Figura 2-8.- Hello OSPF

Los ruteadores adyacentes pasan por una secuencia de estados, y deben estar en su estado completo antes de crear las tablas de enrutamiento y direccionar el tráfico. Cada elemento de enrutamiento envía publicaciones de estado de enlace (LSA) en paquetes de actualización del estado de enlace (LSU). Esas LSAs describen todos los enlaces de los ruteadores quienes al recibirlas de sus vecinos las registran en la base de datos del estado de enlace.
Una vez completas las bases cada ruteador utiliza el algoritmo SPF para calcular la ruta con menor costo hacia un destino conocido, luego la información de enrutamiento es mantenida y cuando existe un cambio en el estado de un enlace se produce la inundación notificándose así el cambio en la red.

Existe un intervalo muerto del protocolo Hello (fig. 2-8) que ofrece un mecanismo sencillo para determinar que un vecino adyacente esta desactivado.

2.6 Aplicaciones sobre el Dominio MPLS

Como todas las arquitecturas de Red, MPLS ofrece un conjunto variado de aplicaciones que dan flexibilidad a un gran número de usuarios que a futuro podrían enviar información a través de esta red, y sus aplicativos serán gestionados de la manera más eficaz y eficiente mediante herramientas poderosas de manejo de Calidad de Servicio, Ingeniería de Tráfico y Redes Privadas Virtuales.

2.6.1 VPNs (Virtual Private Networks)

Una red privada virtual (VPN) se construye a base de conexiones realizadas sobre una infraestructura compartida, con funcionalidades de red y de seguridad equivalentes a las que se obtienen con una red privada. El objetivo de las VPNs es el soporte de aplicaciones tanto *intranet* como *extranet,* integrando aplicaciones multimedia de voz, video, datos, etc.… sobre infraestructuras de comunicaciones eficaces y rentables. La seguridad supone aislamiento y ***privada*** indica que el usuario *cree* que posee los enlaces dedicados. MPLS ofrece muchas ventajas en cuanto a este tipo de redes, que son mucho más eficaces y económicas frente a otras soluciones tradicionales.

Las VPNs basadas en MPLS permiten a los proveedores de servicios desarrollar soluciones escalables y construir el establecimiento para ofrecer servicios de valor agregado que justifiquen el por qué estos deberían migrar sus clientes a VPNs MPLS. Dichos servicios incluyen lo siguiente:

- **Servicio sin Conexión *(Connectionless)*:** Una significante ventaja de las VPNs MPLS es que ellas trabajan sin conexión orientada. Para establecer privacidad en un ambiente IP sin conexión, las soluciones VPN imponen un enlace orientado punto a punto cubierto en la red. Al crear una VPN MPLS sin conexión, no se requiere túneles ni encriptación para la privacidad de la red, lo que elimina la complejidad substancial de creación de las mismas.

- **Servicio Centralizado:** El construir privacidad en la capa de red permite la entrega de servicios dirigidos a un grupo de usuarios representados por una VPN, la cual debe dar a los proveedores de servicio mas de una

forma, que permita conectar privadamente a estos usuarios en una intranet. Dado que las MPLS VPNs son vistas como intranets privadas, se puede utilizar servicios de IP tales como *Multicast* y Calidad de Servicio.

- **Escalabilidad:** Las VPNs construidas basándose en MPLS utilizan el modelo de Pares *(Peers)* y arquitectura sin conexión orientada de capa de red para lograr una solución que puede tender al crecimiento. El modelo de pares requiere que un sitio de un cliente se conecte a un solo ruteador de frontera del proveedor (PE) como opuesto a los otros ruteadores del cliente (CE) en sitios remotos que son miembros de la VPN.

- **Seguridad:** Las VPNs MPLS ofrecen el mismo nivel de seguridad que las VPNs de conexión orientada. Los paquetes de una VPN no viajan inadvertidamente hacia otra VPN.

    1. En el ingreso a la red de un proveedor, se asegura que los paquetes recibidos desde un cliente sean ubicados en la correcta VPN.

    2. En el Backbone se asegura que el tráfico VPN se mantenga separado. Los intentos maliciosos para ganar acceso a la red *(Spoofing)* son tempranamente evitados porque los paquetes recibidos desde los clientes son paquetes IP. Estos paquetes IP deben ser recibidos en una interfaz o subinterfaz en particular que será únicamente identificada con una etiqueta VPN.

- **Facilidad de Creación:** Dado que las VPNs MPLS son sin conexión directa, ningún mapa específico de conexión punto a punto o topologías son requeridas. Se pueden añadir sitios a intranets o extranets y formar

grupos de usuarios de una misma entidad. Al gestionar las VPNs de esta manera, ellas habilitan comunidades de algún sitio dado, maximizando la flexibilidad en la construcción de Intranets y Extranets.

- **Direccionamiento Flexible:** Para lograr que un servicio de redes privadas virtuales sea más accesible, los clientes de un proveedor de servicios pueden diseñar su propio plan de direccionamiento, independiente de los planes de direccionamiento para otros clientes. Algunos clientes utilizan espacios de direccionamiento privado y no desean invertir el tiempo en gastar en la conversión a espacios de direcciones públicas para habilitar la conectividad intranet. Las VPNS MPLS permiten que los clientes continúen utilizando su actual espacio de direcciones sin necesidad de realizar NAT por la provisión de una vista pública o privada de la dirección. Esto pone a disposición que los clientes utilicen sus espacios de direcciones privadas no registradas, y comuniquen libremente por medio de una red pública.

- **Soporte de Calidad de Servicio Integrado:** La Calidad de Servicio es un importante requerimiento para algunos clientes de VPNs IP. Esto proporciona la capacidad para direccionar dos requerimientos VPN fundamentales:

  1. Desempeño predecible e implementación de políticas.

  2. Soporte para múltiples niveles de servicio en una VPN MPLS. El tráfico de red es clasificado y etiquetado en la frontera de la red antes de que sea agregado de acuerdo a políticas definidas por los suscriptores e implementadas por el proveedor y transportadas a través del núcleo del mismo. El tráfico en la frontera de la red puede

ser entonces diferenciado en varias clases por la probabilidad de descarte, por el retardo, por el protocolo de aplicación, etc.

- **Migración Directa:** Para que los proveedores de servicio desarrollen rápidamente los servicios de VPN, utilizan una ruta de migración directa. Las VPNs MPLS son únicas porque se las puede construir sobre múltiples arquitecturas de redes. La migración para el cliente final está simplificada ya que no existe requerimiento para soportar MPLS en el ruteador que posee (CE), y ningunas modificaciones son requeridas para su intranet.

Figura 2-9.- VPNs en MPLS

Como se observa (fig. 2-9), una VPN contiene dispositivos de cliente ligados a los CEs. Estos dispositivos del cliente usan VPNs para intercambiar información entre ellos. Cabe recalcar que solo los PEs tienen conocimientos de las VPNs.

2.6.2 QoS (Quality of Service)

La habilidad para que una red proporcione servicios mejorados para el tráfico seleccionado, sobre varias tecnologías subyacentes de la capa de red (Redes Ruteadas IP, ATM, Frame Relay, etc) se define como Calidad de Servicio (QoS). En particular, las características de QoS proporcionan servicios de redes mejorados y predecibles como los siguientes:

- Soporte de Ancho de Banda Dedicado

- Mejora en características de perdidas

- Prevención y Gestión de la congestión

- Formación del trafico de red

- Configuración de prioridades al tráfico que atraviesa la red.

Del uso de la Calidad de Servicio pueden beneficiarse todas las redes que deseen eficiencia óptima, sin importar que estas pertenezcan a pequeñas, medianas o grandes empresas que tienen sus propios requerimientos de QoS.

Las redes empresariales deben proporcionar soluciones con calidad de servicio de extremo a extremo (usuarios) a través de diferentes plataformas que son partícipes de ellas, proporcionando soluciones para plataformas mixtas que a menudo requieren que se tomen diferentes aproximaciones en lo que concierne a implementación de QoS para cada tecnología utilizada en una red heterogénea. Un claro ejemplo son las empresas que hoy en día

manejan aplicaciones de datos de misión crítica y experimentan elevado tráfico multimedia Web, sintiendo la necesidad de que los equipos que ofrecen QoS prioricen el tráfico asegurando que se obtenga el nivel de servicio que cada aplicación requiere.

La ventaja que se tiene al implementar Calidad de Servicio para controlar las aplicaciones y tipos de tráfico en redes, se refleja en las siguientes características:

- **Control sobre Recursos:** Se tiene control sobre recursos utilizados (ancho de banda, equipamiento, etc.…). Se puede limitar el consumo de ancho de banda sobre un enlace de Backbone para lo que es FTP, o también se puede dar prioridad al acceder a una importante base de datos.

- **Servicios Diferenciados:** Un proveedor de servicios que controla y proporciona QoS puede gestionar los servicios de los clientes diferenciándolos en niveles o en clases.

- **Coexistencia de Aplicaciones de Misión Crítica:** Las características de QoS hacen posibles las siguientes condiciones:

    1. Que el enlace entre el cliente y el proveedor (WAN) sea utilizado eficientemente por las aplicaciones de misión crítica que son muy importantes para los negocios

    2. Que el ancho de banda y mínimo retardo requerido por aplicaciones de voz y multimedia sensible al tiempo de respuesta estén disponibles.

3. Que las otras aplicaciones utilizando el enlace, obtengan sus servicios justos sin interferir con el tráfico de misión crítica.

2.6.3 Traffic Engineering

Ingeniería de Trafico es el proceso de controlar como fluye el tráfico a través de una red utilizando optimización de recursos y el desempeño de la red. Se preocupa básicamente de dos problemas que provienen de los protocolos de enrutamiento, los cuales solamente utilizan la ruta más corta como parámetro cuando construyen una tabla de enrutamiento. La ruta mas corta desde las diferentes fuentes se sobrepone en algunos enlaces, causando la congestión de los mismos. El tráfico desde una fuente hacia un destino excede la capacidad de la ruta más corta mientras una ruta más larga esta siendo poco utilizada.

MPLS puede ser usado como herramienta de ingeniería de tráfico, para direccionarlo dentro de una red en una forma más eficiente de lo que lo hace el enrutamiento original de la ruta mas corta en IP. También puede ser utilizado para controlar aquellas rutas de tráfico que viajan a través de la red y usan más eficientemente los recursos que pueden alcanzarse. Las rutas pueden ser reservadas para el tráfico que es sensible así como para enlaces que son más seguros.

Entre los beneficios que ofrece la Ingeniería de Trafico MPLS se tienen:

- **Alta recuperación de la inversión de la infraestructura de red.** Específicamente, la mejor ruta entre dos equipos de red es determinada tomando en cuenta los parámetros del Backbone y la carga total de tráfico sobre este.

- **Reducción en costos de operación.** Los costos son reducidos porque un numero de procesos importantes son automatizados, incluyendo la configuración, el mapeo y la selección rutas de Ingeniería de Trafico MPLS (Túneles)

2.7 Resumen de la Descripción Funcional de una Red MPLS

La tecnología MPLS centra su funcionamiento principalmente en los Planos de Control y Planos de Envío de los ruteadores, ya sean estos de frontera o del núcleo de la red (LSR o LER). En la componente de control se encuentran los protocolos de enrutamiento y de distribución de etiquetas los mismos que permiten que se comparta la información con dispositivos adyacentes. Al intercambiar información entre los ruteadores, se crea en la componente de envío lo que son las tablas de envío de etiquetas, y las tablas de envío IP (si se trata de LER).

Al ingresar un paquete IP a una Red MPLS, el mismo es etiquetado en la frontera del dominio y viaja a través del backbone según le indiquen los LSRs de tránsito que se encargan de intercambiar las etiquetas hasta que un paquete alcance su destino final donde se toman decisiones dependiendo el siguiente salto correspondiente al paquete, el cual puede ser un salto a otra red MPLS como también IP en el cual se procede a remover la etiqueta del paquete y utilizar enrutamiento convencional. La manera en la que los ruteadores de un dominio MPLS intercambian sus etiquetas puede ser realizada mediante técnicas de solicitud de etiquetas (Downstream bajo demanda o sin solicitar), para lo cual son partícipes los protocolos de intercambio de etiquetas y señalización como LDP y RSVP, los mismos que ayudan también a marcar el camino que un paquete debe seguir.

En MPLS, al igual que en las tecnologías de transporte de TCP / IP, se utilizan los mismos protocolos de enrutamiento dinámico, ya sea vector distancia o estado de enlace (RIP, IGRP, OSPF, BGP); los cuales permiten que el cálculo de las mejores rutas ayuden en las mejoras del desempeño de una red, permitiendo así rápida convergencia, escalabilidad y robustez.

En cuanto a las aplicaciones que ofrece un backbone MPLS se pueden destacar las VPNs, las cuales son la solución más rentable a la hora de enlazar dos sitios geográficamente distantes enviando su información a través de un medio público; la Calidad de Servicio – QoS, la misma que puede ser implementada para clasificar el tráfico, gestionar en ancho de banda según las aplicaciones, y prevenir la congestión enviando el flujo correspondiente a aplicaciones de alta importancia a través de túneles de Ingeniería de Tráfico evitando así la carga en los enlaces de la red.

3 DISEÑO DE LA CONEXIÓN WAN DE UNA EMPRESA MEDIANA CON SUS SUCURSALES

3.1 Diseño de la topología de Red

Cuando se trata de diseñar una red, se deben considerar aspectos importantes ya que al momento de implementar es necesario asegurarse que el diseño escogido fue la topología mas adecuada con los equipos de buen desempeño que puedan ofrecer, confiabilidad y robustez, a los usuarios que deseen utilizar la red de un proveedor para el transporte de su información. Las consideraciones de diseño para la red presentada (fig. 3-1) se hacen pensando en una red mallada completa *(Full mesh),* construida con equipos de la plataforma Cisco 7200, de tal forma que un proveedor con un diseño *full mesh* en su núcleo de red pueda ofrecer garantías de envió de información y que además cuente con caminos adicionales y redundantes según se de el caso de que falle algún nodo en el núcleo de la nube.

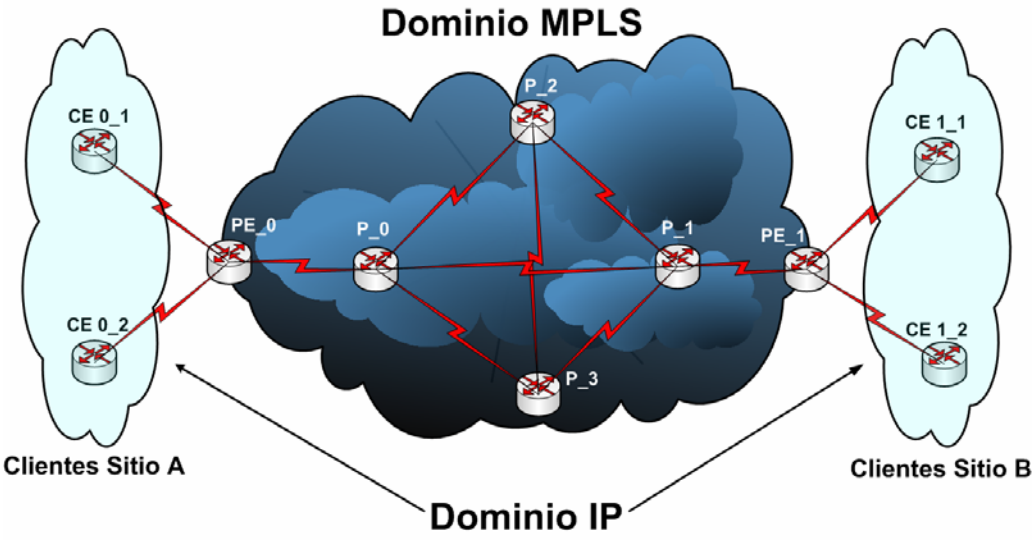


Figura 3-1.- Dominio MPLS y topología mallada completa

Para los usuarios que deseen enviar su información a través de la nube MPLS se debe ofrecer una topología lógica de tal manera que haga parecer que se cuenta con enlaces dedicados desde un sitio matriz hacia cada una de las agencias con las que pueda contar una empresa típica. Una red diseñada con MPLS que será utilizada para el envío de información debe ser transparente a los clientes dado que, al contar con una topología definida se puede alcanzar los destinos por diferentes caminos y así hacer del transporte de la información algo confiable. La figura 3-2 da una referencia de cómo se conectan las agencias de una empresa típica con su matriz para ser servidas de las distintas aplicaciones que deban utilizarse tales como: correo, transferencia de archivos, voz, mensajería instantánea, aplicaciones transaccionales, etc.

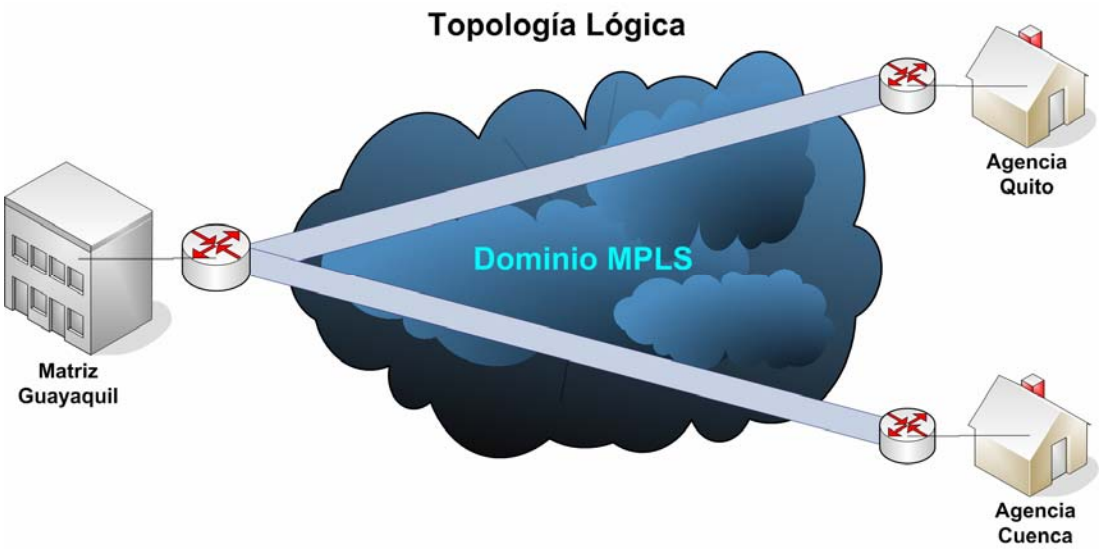


Figura 3-2.- Topología que define la conectividad de una empresa Matriz con sus Agencias

Según la gráfica (fig.3-2) se puede notar que la matriz de una empresa (matriz Guayaquil) puede ofrecer aplicaciones a sus agencias por medio de un dominio publico MPLS que un proveedor de servicios ofrece, buscando como solución mas económica la implementación de VPNs de capa 3 de tal manera que se piense que estos canales VPN son enlaces dedicados.

3.1.1 Topología Full Mesh

Como se puede observar (fig. 3-1), el arreglo de los equipos de red de un proveedor es una topología de malla completa en el *Core.* Este tipo de arreglo *(full mesh)* se caracteriza por tener todos sus nodos conectados entre si para el intercambio de la información. Dada su gran cantidad de redundancias en lo que se refiere a los enlaces, este tipo de topologías es usual en los *Backbones* de los proveedores. Al existir la redundancia de enlaces se garantiza una estructura de *Backbone* confiable, capaz de manejar grandes cantidades de información.

Para el diseño de una red MPLS de un proveedor, es necesario siempre estudiar la posibilidad de crear un núcleo mallado completo, dado que pueden existir situaciones en las que se presenten problemas en algún enlace y es necesario contar con respaldos que reemplacen a la ruta principal. Además se tiene que considerar este tipo de topologías para atender los requerimientos de usuarios que demandan gran uso de ancho de banda mediante el uso de aplicativos que así lo requieran (voz, video, datos) y que mediante la existencia de enlaces redundantes la información pueda ser direccionada por diferentes caminos según permita la Calidad de Servicio e Ingeniería de Trafico aplicada en la red del proveedor.

3.1.1.1 Herramientas representativas en el diseño de la Red MPLS

Como se menciona en capítulos anteriores, una red MPLS esta constituida por elementos de núcleo (*P – Routers* o LSRs) y elementos de frontera (*PE – Routers* o LERs). En lo correspondiente, tanto a la frontera como al núcleo MPLS se toma en cuenta el uso de ruteadores de la serie Cisco 7200, específicamente la plataforma Cisco 7206VXR con un sistema operativo *IOS - c7200-js-mz.124-13b* para uso de proveedores de servicio, el mismo que cuenta con las funcionalidades necesarias, cualidades de procesamiento y gestión de recursos adecuadas a usarse en una red MPLS

Las Características, funcionalidades y aplicaciones soportadas en los equipos de la red MPLS serán detalladas en la siguiente sección.

3.1.1.1.1 Routers Cisco 7200

La serie de ruteadores Cisco 7200 es muy usual para las implementaciones de *Backbones* de redes MPLS. Para este diseño (fig. 3-1) se escoge el ruteador Cisco 7206VXR, el cual soporta enrutamiento multiprotocolo, capacidades Gigabit para mejorar la integración de datos, voz y video en ambientes, tanto empresariales como de proveedores de servicios sobre una amplia variedad de tipos de interfaces ya sean LAN o WAN. Además cuenta con motores de servicios de red para altas velocidades *(NSE – Network Service Engine),* o con motores de procesamiento de red para altas velocidades *(NPE – Network Processing Engine).*

3.1.1.1.1.1 Características Técnicas

Entre las principales características técnicas del Cisco 7206VXR se pueden citar que, las interfaces de red residen en adaptadores de puertos que proporcionan la conectividad entre los tres buses PCI del ruteador y las redes externas. El Cisco 7206VXR posee seis ranuras (numeradas del 1 al 6) para los adaptadores de puerto, una ranura para un controlador de entrada/salida y una ranura para el motor de procesamiento de red. Los adaptadores de puerto se pueden ubicar en cualquiera de las seis ranuras disponibles. Este equipo puede recibir alimentación eléctrica de corriente continua DC, así como de corriente alterna AC, pero no debe ser alimentado con la mezcla de ambas corrientes.

En su aspecto físico (fig. 3-3. Vista Frontal) se puede visualizar cada uno de los componentes del ruteador Cisco 7206VXR.

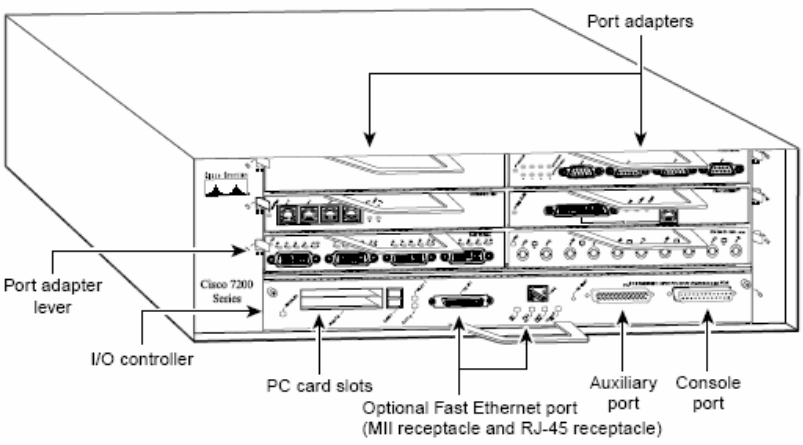


Figura 3-3. - Ruteador Cisco 7206VXR

3.1.1.1.1.2 Funcionalidades.

En esta sección se describe el esquema de numeración y direccionamiento de los adaptadores de puerto para el ruteador, las funciones de reportes y monitoreo del entorno y la remoción e inserción en línea *(OIR – Online Insertion and Removal)*, lo cual ayuda a familiarizarse con las capacidades del dispositivo.

Ranura de Adaptador de Puerto y Numeración de Interfaz Lógica

En el Cisco 7206, el número de ranura del adaptador de puerto es la ranura del chasis en la cual un adaptador de puerto o adaptador de servicio esta instalado, en cambio, el número de interfaz lógica es la localización física del puerto de interfaz en un adaptador de puerto (los adaptadores de servicio no tienen puertos de interfaz). Las ranuras del adaptador de puerto están enumeradas desde el uno hasta el seis; la ranura cero del adaptador de puerto es reservado para el puerto opcional *Fast Ethernet* en la controladora de Entrada/Salida en caso de que sea necesaria. El número de interfaces lógicas depende del tipo de adaptador de puerto.

La dirección física de control de acceso al medio (MAC) o dirección de hardware, es una dirección de capa de enlace de datos estandarizada que es requerida para ciertos tipos de interfaces de red. Estas direcciones no son utilizadas por otros dispositivos en la red ya que ellas son específicas y únicas en cada puerto. El ruteador Cisco 7206VXR emplea un método para asignar y controlar las direcciones MAC de sus adaptadores de puerto, el cual se detalla en la sección correspondiente a direcciones MAC.

Las ranuras de los adaptadores de puerto mantienen el mismo número sin importar que otros adaptadores de puerto o adaptadores de servicio sean instalados o removidos. Sin embargo, cuando se mueve un adaptador de puerto a una ranura diferente, obviamente el número de ranura del adaptador de puerto cambia reflejando la nueva posición.

Se puede identificar los adaptadores de puerto y su ubicación gracias al uso de comandos que muestran información de los mismos que puedan estar incluidos en el Cisco 7206VXR.

Direcciones MAC

Todas las interfaces (puertos) requieren de una dirección única, también conocida como dirección de hardware. Típicamente, la dirección MAC de una interfaz es almacenada en un componente de memoria que reside en la circuitería de la misma; sin embargo, la característica de Inserción y Remoción en línea (OIR) necesita de un método diferente.

La característica OIR permite que se remueva un adaptador de puerto o de servicio y se reemplace éste por otro idénticamente configurado. Si el nuevo dispositivo es reconocido, entonces el sistema inmediatamente lo pone en funcionamiento. Para habilitar la característica OIR, un asignador de

dirección con una única MAC es almacenado en una EEPROM en el plano medio *(midplane)* del ruteador. Cada dirección física es reservada para un puerto y ranura, y las mismas se asignan siguiendo un orden, por ejemplo: para la ranura, cero se asigna la primera dirección MAC y para la ranura seis, se asigna la última dirección MAC. Este esquema de direccionamiento permite que se remueva adaptadores de puerto o de servicio, y que se inserte en otro ruteador sin causar que la dirección física se mueva alrededor de la red o sea asignada a múltiples dispositivos.
Cabe recalcar que si la MAC estuviera almacenada en cada adaptador de puerto, la característica OIR no funcionaría ya que nunca se podría reemplazar algún adaptador por otro idéntico.

Inserción y Remoción en Línea (OIR)

Todos los adaptadores de puerto en el Cisco 7206VXR pueden ser insertados o removidos mientras el dispositivo está en funcionamiento, lo cual permite que se instalen y reemplacen adaptadores sin necesidad de desactivar el ruteador y desactivar las interfaces por medio del software. Cuando se inserta o se remueve un dispositivo en el Cisco 7206VXR, los pines a los cuales se conecta el mismo dentro del ruteador envían señales para notificar al sistema, realizándose así lo siguiente:

1. Escaneo rápido en los cambios de la configuración interna *(Midplane).*

2. Una vez que se inicializa el sistema los nuevos adaptadores insertados pasan inmediatamente a estado de deshabilitados administrativamente *(Administratively down).*

3. Todas las interfaces previamente configuradas en el adaptador de puerto pasan al estado que tenían cuando fueron removidas, es decir que si un

nuevo adaptador similar a uno desinstalado se inserta y si algunas interfaces estaban habilitadas, entonces en el nuevo adaptador también permanecerán en línea.

Ambientes de Monitoreo y Funciones de Reporte

Para propósitos de funciones y ambientes de monitoreo, el motor de procesamiento del ruteador controla y permite que se mantenga normalmente la operación del sistema identificando y resolviendo condiciones previniendo la disminución en desempeño y operación. Las funciones de monitoreo constantemente chequean las temperaturas del aire que circula a través del chasis y las fuentes de voltaje y corriente. Los niveles de temperatura pueden variar, existiendo así valores que pueden ser considerados como umbrales a los que el motor de procesamiento de red en un ruteador Cisco 7206VXR puede trabajar, estos valores se muestran a continuación (tabla 3-1).

Alertas de Alta Temperatura.	Alertas de Temperatura Crítica.	Alertas de Desactivación del Router.
43 ^{o}C	53^{o}C	58^{o}C

Tabla 3-1.- Características de Temperatura del Cisco 7206VXR

Las funciones de reporte periódicamente registran los valores de parámetros medidos, para lo cual se puede recuperar y almacenar esa información que pueda servir de ayuda en análisis posteriores en los que se necesiten de las estadísticas. Además las funciones de reporte muestran avisos en la consola si se da el caso de que alguno de los parámetros monitoreados exceda los umbrales definidos.

3.1.1.1.1.3 Aplicaciones soportadas

Dado que es un ruteador muy usual en los backbones de redes MPLS soporta aplicaciones muy relacionadas con el manejo de grandes cantidades de información tales como: enrutamiento IP – MPLS, Calidad de Servicio (QoS), Redes Privadas Virtuales (VPN), Ingeniería de Tráfico, Túneles, Seguridad en la Red,

3.1.1.2 Herramientas utilizadas por el cliente para la conexión a la red MPLS.

Para los sitios del cliente se ofrece conectividad a la red MPLS de un proveedor mediante ruteadores de la serie Cisco 3700, específicamente la plataforma Cisco 3745 con sistema operativo *IOS - c3745-entservicesk9-mz.124-13b*, el cual cuenta con herramientas de procesamiento adecuadas para el manejo y envío de grandes cantidades de información por parte de los usuarios.

3.1.1.2.1 Routers Cisco 3745.

La serie de los dispositivos Cisco 3700 es una familia de ruteadores modulares que habilitan flexibilidad y desarrollo escalable en las redes, dando a conocer un alto rendimiento en el enrutamiento, seguridad en las aplicaciones, etc.

3.1.1.2.1.1 Características Técnicas.

Estos dispositivos de capa de red proporcionan una conectividad LAN y WAN, nuevos módulos de servicio de alta densidad (HDSM), y soporte para múltiples Módulos de Integración Avanzados (AIMs) con los que permiten entregar los más altos niveles de servicio para las empresas de hoy en día.

Las plataformas de los ruteadores serie Cisco 3745 poseen: dos puertos integrados LAN 10/100, dos ranuras integradas para Módulos de Integración Avanzado (AIM), tres ranuras integradas para tarjetas de interfaces WAN *(WIC – WAN Interface Card),* cuatro ranuras para módulos de red *(NM – Network Module),* dos ranuras para Modulo de Servicio de Alta Densidad *(HDSM – High Density Service Module)* , 32 MB de memoria Flash y 256 MB de DRAM , dos módulos SDRAM (SODIMM) de 128 MB, soporte para la mayoría de protocolos WAN: Frame Relay, PPP, ISDN, X.25, ATM, T1/E1 fraccionado, T1/E1, xDSL, T3/E3, HSSI, soporte para Módulos de Red (NM), soporte para tarjeta de interfaz WAN (WIC) de las series Cisco 1700, 2600 y 3600; tres chasis de Rack montable *(RU – Rack-mountable),* Fuente de Poder Universal DC de 24V o 60V; entres las características adicionales tenemos: Un campo reemplazable para la Tarjeta Madre, una Tarjeta de Entrada/Salida y una Bandeja para el Disipador; *Backplane* pasivo, Fuentes de poder redundantes opcionales AC y DC, Inserción y Remoción en línea de Módulos de Red y fuentes de poder.

3.1.1.2.1.2 Funcionalidades.

Se pueden citar entre las funcionalidades claves de las plataformas Cisco 3745 usadas para empresas y la conexión con sus sucursales a través de una infraestructura publica, características de alto requerimiento en el

desarrollo de aplicaciones avanzadas como son: voz, seguridad, Calidad de Servicio, aceleración de contenidos y entrega, y una alta disponibilidad en la frontera de la red para integrar funciones previamente direccionadas para una combinación de plataformas entre distintos entornos de tecnología. Otras funciones importantes de los ruteadores Cisco 3745, son las características de seguridad de VPNs que ofrecen a los clientes, el sistema de prevención de Intrusos *(IPS – Intrusion Prevention System)* y cortafuegos *Firewalls*. Además estas plataformas ofrecen una infraestructura capaz de transportar tráfico de voz, video y datos a través de túneles.

3.1.1.2.1.3 Aplicaciones Soportadas.

Como ya se menciono anteriormente estos dispositivos soportan aplicaciones de Calidad de Servicio como voz, video y datos a través de túneles, VPNs, etc. A más de ser usuales para transportar la información de los puntos finales de los clientes y de trabajar como CEs, estos dispositivos pueden también usarse para el transporte de información dentro de un *Backbone*

3.2 Resumen Referente a la Descripción del Diseño y Topología de la Red

Para el diseño de una red MPLS, un proveedor de servicios de red debe considerar una serie de aspectos para asegurarse que el diseño a elegir pueda soportar el manejo de grandes cantidades de información y pueda ofrecer seguridad en el transporte de la misma. Consideraciones como: tener un núcleo de red completamente mallado, es decir con todos los dispositivos conectados entre si para tener redundancia de enlaces, es un factor muy importante ya que la información que atraviesa un *backbone* debe tener una serie de caminos opcionales para alcanzar su destino, ya que si algún enlace o algún nodo falla, la información puede ser direccionada por un enlace de respaldo.

En el diseño debe considerarse además la plataforma de dispositivos que deberían utilizarse, ya que aquellos dispositivos tienen que soportar los grandes volúmenes de información, procesando la misma de una manera eficiente y sin ocasionar pérdidas, dando a entender de esa manera que se puede contar con una red fiable, robusta y capaz de entregar la información eficientemente ofreciendo las garantías que se requieren en el transporte. Los dispositivos deben ofrecer la calidad de servicio adecuada según lo demande cierta aplicación ya sea esta de alta prioridad o de prioridad media, ofreciendo además el encaminamiento correcto evitando de esa manera saturar los caminos de la red.

4 VALIDACIÓN E IMPLEMENTACIÓN DEL DISEÑO DE LA RED MPLS MEDIANTE EL SIMULADOR DYNAMIPS

Para un mayor conocimiento y validación de la topología elegida como modelo de la red MPLS es necesario el uso de herramientas de aplicación que simulen el comportamiento de la red interaccionando con entidades o protocolos de tal manera que se pueda estudiar el comportamiento del tráfico que circulará por una nube MPLS. Las características que deben presentar las herramientas de simulación, deben acoplarse a los requerimientos que demande una Red, para que de esa forma las condiciones y problemas que se presenten, se tomen en cuenta en ambientes de Implementación con equipamiento real. La herramienta que se utiliza para este diseño de red MPLS es muy poderosa y se destaca por ser:

a) **Configurable:** De tal manera que se puedan alterar parámetros de red y de tráfico que circula a través de ella.

b) **Rigurosa:** Ya que muestra estabilidad.

c) **Analizable:** Porque los resultados revelan el comportamiento de la red y los posibles problemas que pueden surgir.

d) **Portable:** Dado que es código abierto y fácil de ejecutar en varios sistemas operativos, y se puede dar sustento a posibles colaboraciones en el futuro.

4.1 Introducción al simulador Dynamips

Dynamips es una herramienta de simulación muy poderosa desarrollada inicialmente en Agosto del año 2005 por Greg Anuzelli, quien lo comenzó a implementar como proyecto para simular un ruteador Cisco 7200 en una PC tradicional. Hoy en día gracias a las investigaciones y desarrollos del proyecto, esta herramienta soporta plataformas de las series Cisco 3600 (3620, 3640, 3660), Cisco 3700 (3725, 3745) y Cisco 2600 (2610, 2650XM, 2691), las cuales pueden ser simuladas como si se estuviera trabajando en tiempo real.

Al proyecto Dynamips se han unido colaboradores entre los que se destacan Christopher Phyllot, quien ha desarrollado una herramienta que ayuda a *Dynamips* para que trabaje de tal forma que se optimicen recursos y se mejoren funciones bajo el uso de una sola consola de administración que puede manejar las instancias de los dispositivos Cisco que se estén ejecutando. Esta valiosa herramienta, que permite a *Dynamips* trabajar en un modo conocido como *Hypervisor*, es llamada *Dynagen*.

En las siguientes secciones se presentan completos detalles de la obtención, sistema operativo sobre el cual puede ser montado, la instalación, la utilidad, los entornos y modos en los cuales trabaja, los cuales llevan a Dynamips / Dynagen a ser uno de los simuladores que debe ser usado por todas las comunidades para propósitos de investigación.

4.1.1 Obtención del simulador

Dadas sus condiciones de Portabilidad, lo cual significa que es código abierto *(Open Source)*, *Dynamips* / *Dynagen* puede ser obtenido desde un servidor FTP en la Internet o puede ser descargado desde la dirección que hace referencia al simulador [20].

4.1.2 Instalación del Simulador bajo entorno de Linux

El sistema operativo que ha sido elegido para la instalación de Dynamips ha sido el Sistema Operativo Linux como distribución del mismo, el *CentOS 5*. Así como es sencillo de obtener, *Dynamips* / *Dynagen* es sencillo de instalar, y para ello es necesario colocar el archivo ejecutable del simulador en un directorio del *Linux* disponible para los programas como por ejemplo */opt/dynamips/.* Luego de aquello es necesario que se creen enlaces simbólicos o accesos directos para poder ejecutar el simulador desde la consola sin importar el directorio en el que se encuentre trabajando el usuario de *Dynamips* o *Dynagen*.

Dynamips puede trabajar sólo, o en conjunto con *Dynagen*. La forma de trabajo del simulador se detalla en la siguiente sección mediante guías de aprendizaje del mismo.

4.2 Guía de Aprendizaje para el uso del Simulador

Como toda herramienta de simulación *Dynamips* / *Dynagen* necesita de una guía para que los usuarios se familiaricen con el uso de comandos y directivas que permitan su empleo, para lo cual se necesita primero comprender el objetivo y los alcances de la herramienta, así como las plataformas que soporta.

4.2.1 Objetivos del Simulador

Se ha implementado el simulador con el propósito de que se cumpla con los siguientes usos:

- *Dynamips* debe ser utilizado como plataforma de entrenamiento, con herramientas de software utilizadas en el mundo real, lo cual permita que muchos usuarios se familiaricen con los dispositivos de la plataforma Cisco.

- Probar y experimentar las poderosas y numerosas herramientas del sistema operativo de internetworking (Cisco IOS).

- Chequear rápidamente configuraciones, que puedan ser utilizadas a futuro, en implementaciones con ruteadores reales.

4.2.2 Plataformas Soportadas

Como se mencionó anteriormente las plataformas que soporta la herramienta de simulación son variadas pero, para el diseño presentado se hace hincapié en dos versiones de ruteadores Cisco que soporta el simulador y que además hacen referencia a la red MPLS y a los clientes que se conectan a dicha red (fig. 3-1).

4.2.2.1 Plataforma Cisco 7200

Para este tipo de plataformas, que por cierto es muy usual en los Backbones de los proveedores de servicio, *Dynamips* simula adaptadores de puertos como los que se menciona a continuación (tabla 4-1):

Tipo de Adaptador	***Código***	***Descripción***
Tarjetas Gigabit Ethernet	C7200 - IO – GE - E	Giga Ethernet, 1 puerto (ranura 0 solamente)
Tarjetas Fast Ethernet	C7200 - IO – FE	Fast Ethernet, 1 puerto (ranura 0 solamente)
	C7200 - IO – 2FE	Fast Ethernet, 2 puertos (ranura 0 solamente)
	PA - FE – TX	Fast Ethernet, 1 puerto (ranura 1 - 6)
	PA - 2FE – TX	Fast Ethernet, 2 puertos (ranura 1 - 6)
Tarjetas Ethernet	PA - 4E	Ethernet, 4 puertos (ranura 1 - 6)
	PA - 8E	Ethernet, 8 puertos (ranura 1 - 6)
Tarjeta ATM	PA - A1	ATM, 1 puerto (ranura 1 - 6)
Tarjetas Seriales	PA - 4T+	Serial, 4 puertos (ranura 1 - 6)
	PA - 8T	Serial, 8 puertos (ranura 1 - 6)

Tabla 4-1.- Adaptadores de Puertos para la plataforma Cisco 7200

Nota: Como se especifica en la tabla 4-1, la ranura cero solo debe ser utilizada para las tarjetas antes mencionadas.

4.2.2.2 Plataforma Cisco 3745

Para propósitos de simulación de clientes se emplea el ruteador Cisco 3745 como ya se mencionó. Para este dispositivo, *Dynamips* emula los siguientes módulos de red:

Tipo de Modulo de Red	*Codigo*	*Descripcion*
Tarjetas Fast Ethernet	NM - 1FE – TX	Fast Ethernet, 1 puerto
Modulo Switch Ethernet	NM - 16ESW	Módulo Switch Ethernet, 16 puertos
Tarjeta Serial	NM - 4T	Serial, 4 puertos

Tabla 4-2.- Módulos de red para la plataforma Cisco 3745

4.2.3 Comandos y Directivas para la Instalación de Dynamips

Como es de conocimiento, la obtención de una herramienta de simulación *(Open Source)*, es muy fácil de conseguir en la *web*. *Dynamips* puede ser obtenido en sus distintas versiones y para diferentes sistemas operativos desde un servidor FTP, o desde el sitio *Web* del producto.

Para su instalación es necesario seguir un orden sistemático, comprendiendo así la utilidad de la herramienta y sus alcances. Entre los pasos que se deben seguir a la hora de instalar *Dynamips*, se citan los siguientes:

a) Obtención del Simulador en una versión determinada y para un sistema operativo específico sobre el cual se ejecutará (32bits o 64bits).

b) Desempaquetarlo (en caso que venga comprimido) en un directorio disponible para el uso de programas y crear una carpeta donde será almacenado (para el caso de *Linux* -> recomendable */opt/dynamips*)

c) Abrir una Terminal o Consola, e ir al directorio correspondiente en el cual se encuentra dynamips (en *Linux "cd /opt/dynamips/"*).

d) Una vez que se localice a Dynamips se procede a crear un enlace simbólico o acceso directo para que el simulador pueda ser ejecutado desde cualquier directorio en el que se encuentre ubicado el usuario en la consola (en *Linux*: *"ln –s /opt/dynamips/nombre_de_archivo /usr/local/bin/")*.

e) Luego de creado el acceso directo se procede a la obtención del sistema operativo de la plataforma del ruteador que se desea simular, respetando las plataformas soportadas.

f) Cuando se obtiene la imagen del sistema operativo Cisco IOS (por defecto comprimido), se procede al desempaquetamiento de la misma no sin antes haber almacenado la imagen en un directorio del sistema operativo *(ej. /opt/images/).* Luego es desempaquetada como sigue: *"unzip – p c7200-js-mz.124-13b.bin > image7200.bin".*

g) Finalmente la herramienta está lista para proceder a arrancar el sistema operativo de Cisco, procediendo con las directivas que se detallan en la siguiente sección.

4.2.4 Opciones de la Línea de Comandos de Dynamips

Dynamips posee una variedad de comandos que sirven para su ejecución y además ayudan a optimizar los recursos de la PC donde se ejecuta; como por ejemplo el uso adecuado de recursos de memoria RAM y un buen

desempeño del procesamiento de la maquina. En la tabla 4-3 se muestran los comandos que cumplen las funciones mencionadas anteriormente.

Comando	Parámetro	Descripción
-H	***Puerto &***	Trabaja Dynamips en modo Hypervisor
-P	***Plataforma***	Especifica la plataforma que será ejecutada
-j		Deshabilita el Compilador JIT (Just in Time). Si se deshabilita, el proceso se vuelve muy lento
--exec-area	***Size***	Configura el tamaño del área de ejecución (Por defecto 64MB)
--idle-pc	***Pc***	Configura el valor de carga del procesador (por defecto deshabilitado)
--sparse-mem		Permite el uso de memoria de acuerdo a lo que el Router necesita.
-i	***instance***	Setea el ID de una Instancia (Router). Por defecto es cero.
-r	***Ram_size***	Setea el tamaño de la RAM virtual
-g	***ghost_ram***	Crea un archivo de memoria RAM
-G	***ghost_ram***	Utiliza el archivo de memoria RAM
-o	***Rom_size***	Setea el tamaño de ROM virtual
-n	***nvram_size***	Setea el tamaño de la NVRAM
-c	***Conf_reg***	Configura el registro de configuración
-m	***Mac_addr***	Setea la dirección MAC del chasis. Por defecto se genera automáticamente.
-C	***cfg_file***	Importa una configuración IOS en la NVRAM
-X		Permite al simulador que trabaje sin archivo de memoria RAM
-R	***Rom_file***	Carga una ROM alterna. Por defecto viene inmersa
-k	***clock_div***	Setea el divisor de Reloj (Por defecto 4)
-A	***Port***	AUX esta en un puerto TCP
-B	***si_desc***	AUX esta en una interfaz serial
-a	***cfg_file***	Archivo de configuración de un switch virtual ATM
-f	***cfg_file***	Archivo de configuración de un switch virtual Frame Relay
-E	***cfg_file***	Archivo de configuración de un switch Ethernet
-b	***cfg_ file***	Archivo de configuración de un bridge virtual.
-e		Muestra una Lista de dispositivos de red de la máquina host.

Tabla 4-3.- Comandos para Dynamips

Detalle de obtención de valores de procesamiento.

--idle-pc <idle_pc>: Permite ejecutar una instancia de ruteador sin cargar el CPU al 100%. Esto implica que puede simular un largo número de instancias por maquina real. Para determinar "idle-pc" se deja arrancar normalmente el simulador con el cisco IOS y una configuración IOS vacía. Cuando la imagen haya cargado completamente, ingresamos al modo privilegiado y presionamos la combinación de teclas *"Ctrl -] + i"* y el siguiente mensaje nos ha de salir: *"Please wait while gathering statistics".* Al final tendremos muchos valores para *"--idle-pc"*. Se puede tratar con algunos valores antes de encontrar el mejor.

Detalle del modo Hypervisor

-H <puerto &>: Para trabajar *Dynamips* como modo *Hypervisor* se debe crear un archivo de configuración el cual permite que se administre todas las instancias pertenecientes a los ruteadores de una topología de red en una única consola general.

4.2.5 Especificación de Opciones para la plataforma Cisco 7200.

Por defecto *Dynamips* ha sido creado para trabajar con la plataforma Cisco 7200. La tabla 4-4 especifica las directivas para elección del tipo de chasis, inserción de adaptadores de puerto y conexiones para las interfaces de los dispositivos de Red.

Comando	Parámetro	Descripción
-t	***npe_type***	Selecciona el tipo npe (chasis). Por defecto npe-200
-M	***mid_plane***	Selecciona el plano medio ("std" o "vxr")
-p	***pa_desc***	Define un adaptador de puerto
-s	***pa_nio***	Añade una interfaz IO de red a un adaptador de puerto

Tabla 4-4.- Opciones para la plataforma Cisco 7200

4.2.6 Especificación de Opciones para la plataforma Cisco 3745.

Para la plataforma Cisco 3745 es necesario especificar el comando “-P” y luego de aquello poder utilizar las opciones que se describen en la tabla 4-5, y trabajar con este tipo de dispositivos de red, tal y cual se estuviera manipulando un ruteador real.

Comando	Parámetro	Descripción
-p	***nm_desc***	Define un modulo de Red
-s	***nm_nio***	Vincula una Interfase de Entrada/Salida de Red al modulo de Red

Tabla 4-5.- Opciones para la plataforma Cisco 3745

4.3 El Cisco IOS (Internetworking Operative System)

Una de las principales razones por la que Cisco es la empresa de redes numero uno en el mercado, es debido a su *Sistema Operativo de Internetwork (IOS).* El IOS provee una función similar a Microsoft Windows XP o Linux; este controla y administra el hardware en el que esta ejecutándose. Básicamente, el IOS provee la interacción entre la maquina y

la persona, habilitando así la ejecución de comandos para configurar y administrar el dispositivo Cisco.

4.3.1 Introducción al Cisco IOS

Originalmente el IOS fue desarrollado para ruteadores Cisco, pero en los últimos años, Cisco ha sido el portador de IOS para otras plataformas, incluyendo los *switches Catalyst.* Cisco ha empleado muchos años mejorando el IOS, tanto como para añadir características y nuevas tecnologías que son introducidas en el mercado. Entre las ventajas del IOS se incluyen:

Características: El IOS presenta un amplio arreglo de características para protocolos y funciones que proveen conectividad, escalabilidad, fiabilidad y soluciones de seguridad para infraestructuras de red de cualquier tamaño.

Conectividad: Soporta una variedad de tecnologías en la capa de enlace de datos para ambientes LAN y WAN, incluyendo cableado de cobre, fibra, y también medios inalámbricos.

Escalabilidad: Soporta plataformas con chasis fijos y modulares, dejando a disposición que se adquiera el hardware apropiado según los requerimientos del usuario.

Fiabilidad: Para asegurar que los recursos críticos siempre sean alcanzables, Cisco ha desarrollado algunos productos y herramientas al IOS para proporcionar redundancia a una red.

Seguridad: Con el IOS, se puede controlar estrictamente el acceso a una red y dispositivos de la misma en concordancia con las políticas de seguridad interna asignadas.

Actualmente existen muchos métodos de acceso a un dispositivo Cisco, entre los que se incluyen: puerto de consola, puerto auxiliar, Terminal virtual (Telnet), explorador Web y estaciones de gestión SNMP. Una interfaz de consola proporciona conexión de acceso serial hacia un ruteador, donde se pueden ingresar los comandos basándose en modo texto. Para acceder a un dispositivo Cisco desde una estación remota es necesario crear una configuración básica en la cual se incluye el direccionamiento IP. Además, para desarrollar la configuración inicial es necesario acceder al puerto de consola del dispositivo Cisco.

El Cisco IOS permite a los dispositivos realizar los siguientes pasos al momento de inicializarse:

1.- Chequeo de Hardware

2.- Localización y carga del Sistema Operativo

3.- Localización y ejecución del archivo de configuración del dispositivo.

El software Cisco IOS utiliza una interfaz de línea de comandos *(CLI)* como entorno de consola tradicional, la misma emplea una estructura jerárquica que requiere el ingreso a distintos modos para realizar tareas particulares, como la configuración de Interfaces, configuración de protocolos de enrutamiento, creación de mapas y políticas de clases, entre otros. Al ingresar a cualquiera de los modos específicos, la petición de entrada del

ruteador cambia para señalar el modo de configuración en uso y solo acepta los comandos que son adecuados en dicho modo.

El IOS suministra un servicio de interprete de comandos, denominado comando ejecutivo *(EXEC)*, el mismo que luego de ingresado un comando, lo valida y lo ejecuta. Como característica de seguridad el Cisco IOS divide las sesiones en dos niveles que son: el modo *EXEC* Usuario y el modo *EXEC* privilegiado. A continuación se dan a conocer las características resaltantes de cada modo.

El modo ***EXEC* Usuario** permite solamente una cantidad limitada de comandos de monitoreo básicos. A menudo se le describe como un modo de visualización únicamente por lo que no permite cambiar la configuración de un dispositivo Cisco. Para conocimiento es necesario tener presente que la petición de entrada para este modo es: **“>”**.

El modo ***EXEC* Privilegiado** da acceso a todos los comandos del dispositivo Cisco. Se puede configurar este modo para que solicite una contraseña del usuario antes de dar acceso. Para mayor protección también se puede configurar para que solicite una ID de usuario lo cual permite que únicamente los usuarios autorizados puedan ingresar al dispositivo. Los comandos de configuración y administración requieren que un administrador de red se encuentre en este modo, así como también para el ingreso al modo de configuración global y a los demás modos específicos. Para conocimiento es necesario tener presente que la petición de entrada para este modo es: **“#”**.

Para los sistemas operativos de Internetworking, Cisco suministra imágenes para muchos dispositivos, las mismas que proveen funcionalidades distintas y adecuadas para cada plataforma, los recursos de memoria disponibles y

las necesidades de los clientes. El esquema de denominación de las distintas versiones del software Cisco IOS consta de tres partes:

a) La plataforma en la que se ejecuta la imagen.

b) Las características especiales que permite la imagen.

c) El lugar donde se ejecuta la imagen y si la imagen ha sido comprimida en formato ***.zip**.

Las consideraciones importantes al momento de seleccionar una nueva imagen del IOS, es la compatibilidad con las memorias flash y RAM del dispositivo Cisco. Cuanto más reciente sea la versión y cuantas más características brinde, mayor será la cantidad de memoria requerida. El sistema operativo Cisco IOS permite que los dispositivos que lo usan tengan tres entornos o modos de operación, entre ellos están:

a) Monitor de la ROM

b) ROM de arranque

c) Cisco IOS

Los comandos de inicio generalmente se cargan en la RAM y ellos activan uno de estos entornos de operación. Existe un registro de configuración, el cual puede ser usado por un administrador del sistema para controlar el modo de inicialización por defecto que tiene un dispositivo de red.

El monitor de la ROM provee funcionalidad y diagnósticos de bajo nivel. Es muy usual en la reactivación luego de que una falla en el sistema ocurre y

para recuperar una contraseña perdida. A este modo de operación únicamente se ingresa mediante una conexión física directa en el puerto de la consola.

Cuando un dispositivo de red opera en el modo de ROM de arranque, solamente está disponible un subconjunto limitado de la funcionalidad del Cisco IOS. Este modo de operación permite las acciones de lectura y escritura en memoria flash y es utilizado principalmente para reemplazar la imagen del software Cisco IOS que se guarda en esta memoria.

El funcionamiento normal de un dispositivo Cisco requiere el uso de la imagen completa del sistema operativo de Internetwork tal como esta almacenado en la memoria flash. En algunos dispositivos, el IOS se ejecuta directamente desde el flash, sin embargo, la mayoría de los dispositivos (ruteadores) requieren que se cargue una copia del IOS en la RAM, para ser ejecutado desde allí. Cabe recalcar que algunas imágenes del sistema operativo se guardan en el flash en un formato comprimido, y deben expandirse o desempaquetarse al cargarse en la RAM.

Posteriormente se describe algunas de las herramientas de configuración en la interfaz de línea de comandos (CLI) específicamente para MPLS, lo cual ayuda a que las tareas de gestión y configuración se conviertan en fáciles de realizar.

4.3.2 Comandos de Configuración específicamente para MPLS.

Aunque existan diversas imágenes del software de configuración Cisco IOS, para cada modelo y funcionalidad de los dispositivos, la estructura básica de los comandos de configuración es la misma. Las destrezas de configuración

y diagnostico de fallas que se adquieren en cualquiera de los dispositivos, son útiles en una amplia gama de productos.

Dada su flexibilidad, MPLS puede ser habilitado para la entrega de servicios IP sobre cualquier tecnología de transmisión de datos. Sus procedimientos de configuración en el sistema operativo de internetwork ofrecen escalabilidad en una red, de tal manera que los cambios por configuraciones no afecten en su totalidad a una infraestructura ya implementada.

A continuación (tabla 4-6) se detallan los comandos básicos que hacen posible las funcionalidades de envío y control MPLS, los mismos que además son muy usuales en las configuraciones realizadas en la implementación de la red MPLS presentada en capítulos anteriores (fig. 3-1).

Comando	Descripción
ip cef	Habilita de manera global una funcionalidad de envío y conmutación propietaria de Cisco (Requerida)
mpls ip (Configuración global)	Habilita el envío MPLS de paquetes de longitud IPv4, normalmente caminos enrutados por la plataforma (Requerida)
mpls ip (Configuración de interfaz)	Habilita el envío MPLS de paquetes de longitud IPv4, normalmente caminos enrutados para una interfaz en particular (Requerida).
mpls ip default-route	Habilita la distribución de etiquetas asociándola con la ruta IP por defecto.
mpls label range	Configura el rango de etiquetas locales disponibles para el uso en los paquetes.
show mpls forwarding-table	Muestra el contenido de la Base de Información de Envío de Etiquetas (LFIB).
show mpls interfaces	Muestra información acerca de una o más interfases que han sido configuradas para MPLS.
show mpls label range	Muestra el rango de etiquetas locales disponibles para uso de paquetes.
debug mpls adjacency	Muestra cambios en la entrada de conmutación de etiquetas en la base de datos adyacentes.
debug mpls events	Muestra información acerca de eventos MPLS significativos.
debug mpls packets	Muestra paquetes etiquetados conmutados por el router.

Tabla 4-6.- Comandos de configuración y monitoreo básicos para MPLS

Cabe recalcar que existen muchos mas comandos, que permiten la configuración de aplicaciones especificas, ligados con el uso de la tecnología MPLS que se detallan en implementaciones presentadas en secciones posteriores.

4.3.3 Instancias a Simular

Las instancias a simular son aquellos elementos partícipes de la topología MPLS y los elementos del dominio IP que forman parte de la red de los clientes. Estos dispositivos son simulados con la herramienta *Dynamips* trabajando en modo *Hypevisor (Dynagen)* para lo cual se crea un archivo que representa todas las conexiones necesarias entre los elementos pertenecientes a la red MPLS (fig. 3-1).

Tanto los ruteadores de frontera como los ruteadores del núcleo de la red cuentan con funcionalidades de envío MPLS necesarias para ofrecer la seguridad al momento de transmitir la información proveniente de los dispositivos localizados en el lado de los clientes que hacen uso de esta red para comunicarse con sus sitios remotos. Virtualmente se simula los ruteadores de serie Cisco 7206VXR para la nube MPLS y los ruteadores de la serie Cisco 3745 para representar los clientes. Aunque la herramienta de simulación permite ejecutar muchas plataformas más livianas para el caso de los clientes, las elegidas se las ha escogido por el desempeño que poseen al momento de la gestión de información.

4.3.4 Instancias de la Red MPLS

Como es de conocimiento los dispositivos que conforman la red MPLS son los equipos de frontera o *PE – Routers* y los dispositivos del núcleo o los *P – Routers*. En la sección de conexión de las instancias se especifica la creación de las mismas en un archivo de configuración para la topología de Red, el mismo que cumple las funciones de gestionar recursos, conexiones, así como emparejamiento *(matching)* de interfaces LAN de un dispositivo de red con interfaces LAN de una PC.

4.3.4.1 Descripción de las Instancias

Cada uno de los dispositivos simulados es creado en el archivo de configuración haciendo referencia a la imagen o sistema operativo que usará. Las características que tendrá y las herramientas que proporcionará al momento de arrancar completamente, podrán ser configuradas tal como si se estuviera trabajando en un entorno real.

Para los ruteadores cuya ubicación se encuentra en la Frontera y en el núcleo de la red MPLS, se crean en el archivo de configuración dispositivos con los cuales se pueda realizar las funciones necesarias correspondientes a las necesidades y demandas de un *Backbone*. Los Dispositivos Cisco 7206VXR cuentan con una tarjeta serial con 4 puertos *(WIC-4T)* y una tarjeta *Fast Ethernet* con 2 puertos, procesador versátil, memoria RAM de 256MB, y chasis NPE–400. Por otro lado, los dispositivos Cisco 3745 que representan los clientes cuentan con funcionalidades multipropósito y con características entre las que se mencionan: una tarjeta serial de 4 puertos (WIC-4T), una tarjeta *Fast Ethernet* de 2 puertos (uno de los cuales se empareja a una tarjeta *Fast Ethernet* de la PC), una Memoria RAM de 128MB, entre otras.

4.3.4.2 Conexión entre las Instancias

Las conexiones entre los dispositivos de la red MPLS y desde el cliente hacia la misma son muy sencillas mientras se trabaje en modo *Hypervisor*. Dichas uniones entre los dispositivos se realizan dentro del archivo de configuración de la topología de la red y a su vez dan un claro entendimiento a los lectores que observen el archivo en primera instancia.

4.3.4.2.1 Comandos para la Conexión entre Instancias

Para comenzar describiendo las funcionalidades de cada uno de los comandos que deben ser colocados en el archivo de configuración de la red completa, a continuación (tabla 4-7) se muestra la descripción de los comandos utilizados para la representación de la red MPLS (fig. 3-1). En el Anexo A se da el ejemplo del archivo de configuración de la topología de la red.

Comando	Parámetro	Descripción
Ghostios	***true / false***	Permite el uso de memoria compartida
Sparsemem	***true / false***	Permite el uso de memoria que el router necesita en cada fase de trabajo sin desperdiciar recursos
[localhost]		Indica al simulador que se trabaja en una PC local
[[7200]] ó [[3745]]		Indica la plataforma a ser utilizada
Image	***Ruta***	Indica la ubicación de la Imagen del Sistema Operativo (Cisco IOS)
Npe	***npe - <numero>***	Indica el tipo de Chasis que utiliza el Cisco 7200
RAM	***<tamaño>***	Especifica la cantidad de memoria que utiliza la plataforma para ser simulada
[[ROUTER <nombre>]]		Etiqueta del dispositivo. La palabra "ROUTER" puede variar. Se puede especificar un switch "SW".
Model	***<numero de la serie Cisco>***	Es necesario especificar el modelo del dispositivo en caso de que se trate de un Router diferente a la plataforma Cisco 7200 que es aquella que la herramienta de simulación ejecuta por defecto
slot<numero>	***Adaptador***	Especifica el tipo de Adaptador de Puerto que se conecta a una ranura
Interface<slot/port>	***Etiqueta <slot/port>***	Indica una Conexión. Ej: s1/1 = CE1_1 s1/1.
Interface<slot/port>	***NIO:NIC***	Especifica el emparejamiento de la interfaz (solo Fast Ethernet) de un Router a la Tarjeta de Red de la PC. Ej: f0/0 = NIO_linux_eth:eth0

Tabla 4-7.- Descripción de Opciones de Dynamips como Hypervisor

Dado que *Dynamips* también trabaja independientemente sin necesidad de *Dynagen* (modo no *Hypervisor*), existen los comandos que especifican la creación del dispositivo, y las conexiones. Estos son especificados en la misma línea de comandos en la consola que se ejecuta el simulador.

4.3.4.3 Configuración de los Routers de la Red MPLS

Antes de comenzar a configurar dispositivos de red que serán usados en una Topología con direcciones IP definidas, es necesario llevar un orden sistemático de los pasos que se tomarán para que estos dispositivos realicen las funciones de enrutamiento de información, gestionando además recursos dentro del dominio en el que trabajan y mostrando robustez en sus funcionalidades al momento de recibir grandes volúmenes de datos.

En la presente sección se da a conocer los mecanismos para la configuración de enrutamiento y manejo de etiquetas que se dan en una Red IP - MPLS que utiliza herramientas de buen desempeño como lo son: protocolos de enrutamiento de estado de enlace (IGPs), protocolos de Frontera y exteriores (EGPs) y protocolos de distribución de etiquetas.

4.3.4.3.1 Configurando los P - routers (Provider Routers)

Los dispositivos ubicados en el núcleo de la red MPLS como lo son los *P – Routers* (LSRs), son aquellos en los que se gestiona la información y se decide cual es la mejor ruta para alcanzar un destino. En el núcleo de una red deben existir redundancias y los dispositivos que conforman esta parte de la topología deben contar con entidades que le permitan mantener un mapa de las conexiones adyacentes (estado de enlace), a cada uno de ellos.

Estos protocolos de estado de enlace muy conocidos como protocolos de la ruta más corta (SPF) son eficaces a la hora de mantener la información necesaria de cambios en los sistemas y son muy usuales en los *Backbones* de los grandes proveedores de servicio. Al implementar estas herramientas en los dispositivos de red se asegura la estabilidad, flexibilidad, y seguridad a la hora de mantener operativos los enlaces principales y de respaldo que pueda poseer una red.

4.3.4.3.1.1 Los Protocolos de Enrutamiento

En MPLS para lo correspondiente a protocolos de enrutamiento en el núcleo de la red, es muy usual el aplicar el Protocolo de solamente la ruta más corta primero (Only Shortest Path First – OSPF), ya que el mismo obliga a que los dispositivos de red mantengan una coordinación entre ellos mientras se encuentren en una misma área. En la sección de configuración de OSPF de detalla los pasos necesarios para la configuración de esta herramienta de enrutamiento, que es además muy poderosa y eficiente a la hora de reestablecer un enlace que pudo haberse perdido en algún momento.

4.3.4.3.1.1.1 OSPF

El enrutamiento OSPF utiliza el concepto de áreas. Dado que cada ruteador contiene una base de datos completa de los estados de enlace de un área específica, a la misma se le puede asignar un número comprendido entre 0 y 65535. Sin embargo, al área correspondiente al *Backbone* se le suele numerar con el cero porque en caso de que existan numerosas áreas utilizando OSPF, estas deben conectarse a la principal.

La configuración de este protocolo de estado de enlace requiere que se active el proceso en un ruteador con las direcciones de red y la información de área especificada. Las direcciones de red se configuran con una máscara conocida como *wildcard,* más no, con una máscara de subred como se ejecuta en otros tipos de enrutamiento. La máscara *wildcard* representa las direcciones de enlaces o de dispositivos terminales que pueden estar presentes en un segmento específico. Los identificadores de área pueden ser denominados como números enteros o con la notación decimal punteada como las direcciones IP.

El ID de proceso es un número que se utiliza para identificar un proceso de enrutamiento OSPF en un ruteador, en el cual pueden coexistir múltiples procesos que utilicen este protocolo. Es necesario que cada red, subred o dirección de interfaz, al ser añadida a una tabla de enrutamiento OSPF, se le identifique el área al que pertenece. En la tabla 4-8 se especifican los comandos de configuración de OSPF y se describe la utilidad de los mismos.

Comando	Propósito
Router (config)# **router ospf** <*process-id*>	Habilita en enrutamiento OSPF.
Router (config - router)# **network** <*ip-address*> <*wildcard-mask*> **area <***area-id*>	Define una interfaz en la cual se ejecuta OSPF y señala el área para aquella interfaz

Tabla 4-8. - Configuración de Enrutamiento OSPF

4.3.4.3.1.2 Habilitando MPLS

Una vez configurado el encaminamiento de paquetes IP y teniendo presente que se tiene una Red MPLS, es necesario habilitar las funcionalidades multiprotocolo las cuales permitan que a los paquetes IP se les añada etiquetas para el envió MPLS.

Mediante la tabla (4-9) se indican los comandos para la configuración de MPLS en un ruteador Cisco 7206VXR.

Comandos	Propósito
Router(config)# **ip cef**	Habilita de manera global una funcionalidad de envío y conmutación propietaria de Cisco (Requerida)
Router(config)# **mpls ip**	Habilita el envío MPLS de paquetes de longitud IPv4, normalmente caminos enrutados por la plataforma (Requerida)
Router(config)# **interface** <type> <slot/port>	Permite ingreso al modo de configuración de interfaz (Requerida)
Router(config - if)# **mpls ip**	Habilita el envío MPLS de paquetes de longitud IPv4, normalmente caminos enrutados para una interfaz en particular (Requerida).

Tabla 4-9.- Configuración de MPLS

4.3.4.3.1.2.1 Intercambio de Etiquetas - LDP (Label Distribution Protocol)

Para la distribución de etiquetas en las interfaces es necesario especificar el protocolo que se utilizará y que de esa manera los dispositivos vecinos realicen el intercambio correspondiente y la negociación para luego crear las respectivas tablas que indican las etiquetas correspondientes a los paquetes entrantes. El comando presentado (tabla 4-10) detalla la sencilla configuración para la distribución de etiquetas en una interfaz.

Comandos	Propósito
Router(config - if)# **mpls label protocol** <ldp | tdp | both>	Configura el protocolo de distribución de etiquetas en una interfaz

Tabla 4-10. – Configuración de Señalización LDP

Las configuraciones de los dispositivos del núcleo de red MPLS pueden ser vistas desde los *Anexos B-2* al *Anexo B-5.*

4.3.4.3.2 Configurando los PE - Routers (Provider Edge Routers).

Los dispositivos de red ubicados en la frontera MPLS, son aquellos que se encargan de recibir los paquetes y añadir la etiqueta correspondiente a su FEC para que pueda ser enviado a través del dominio. Estos dispositivos trabajan por lo general con varios protocolos de enrutamiento ya que los mismos cumplen funciones como: enrutar paquetes desde y hacia los clientes, enrutar paquetes hacia los ruteadores del núcleo de red o hacia otros dominios por medio de otros dispositivos de frontera.

4.3.4.3.2.1 Los Protocolos de Enrutamiento

Dado que los dispositivos de frontera trabajan en dos ambientes (IP y MPLS), es necesario que estos trabajen con enrutamiento para los dos entornos. Al ser los encargados de recibir un paquete desde un dominio IP ellos deben utilizar un protocolo que les permita la interconexión con un mundo exterior diferente del dominio al que pertenecen. Al etiquetar el paquete IP el siguiente paso es encaminarlo dentro del dominio MPLS, para lo cual debe utilizar una entidad que le permita la comunicación con sus vecinos dentro de la misma red los cuales son los ruteadores del núcleo.

4.3.4.3.2.1.1 OSPF

Este protocolo de estado de enlace es el encargado de establecer la comunicación entre los dispositivos de frontera y los dispositivos del núcleo de red. Su configuración es idéntica a la mostrada en la tabla 4-8 ya que estos ruteadores también forman parte de un área compartida con los ruteadores del núcleo, al tener interfaces conectadas hacia el interior de la red.

4.3.4.3.2.1.2 BGP

El protocolo de gateway de frontera BGP, permite la comunicación entre dominios por lo que su implementación debe ser únicamente en las fronteras de una red. Para la conexión de sitios locales con sitios remotos mediante VPNs, este protocolo es muy usual ya que además permite ser trabajado como protocolo de interiores y se utiliza para la comunicación específica entre dispositivos de frontera. Su implementación debe referirse al uso de sistemas autónomos y dado que en esta implementación se utiliza como BGP interior, el sistema autónomo será único y servirá para identificar a la nube MPLS como un sistema bajo una administración común. La tabla 4-11 muestra en detalle los pasos a seguir a la hora de implementar la comunicación BGP entre dispositivos de frontera que pertenecen a un mismo sistema autónomo (iBGP).

Comandos	Propósito
Router(config)# **router bgp** <*AS número*>	Configura el proceso de enrutamiento IBGP con el número de sistema autónomo que será pasado a otros vecinos IBGP
Router(config-router)# **neighbor** <ip-address | peer-group-name> **remote-as** <AS-number>	Especifica la dirección IP de un vecino con el cual se establecerá en enrutamiento BGP identificando el sistema autónomo al que pertenece.
Router(config-router)# **neighbor** <ip-address | peer-group-name> **update-source** <loopback-interface>	Configura a BGP para que utiliza cualquier interface operacional en conexiones TCP
Router(config-router)# **neighbor** <ip-address | peer-group-name> **activate**	Establece el emparejamiento con un vecino especificado.

Tabla 4-11.- Configuración de Enrutamiento BGP

4.3.4.3.2.2 Habilitando MPLS.

Una vez configurado los protocolos de encaminamiento dinámicos se debe habilitar MPLS al igual que en los ruteadores participantes del núcleo de red, para tener las funcionalidades multiprotocolo y se añada las etiquetas a los paquetes entrantes al dominio MPLS. Los comandos para la configuración MPLS de los ruteadores mencionados, son mostrados en la tabla 4-9.

4.3.4.3.2.2.1 Intercambio de Etiquetas - LDP (Label Distribution Protocol)

Para configurar el manejo de etiquetas en los dispositivos de frontera se hace referencia al comando detallado anteriormente (tabla 4-10) en la sección de Configuración de Intercambio de etiquetas en los *P-Routers*.

Para tener clara la idea de la configuración de los dispositivos de frontera véase los Anexos B-1 y B-6

4.3.4.4 Configuración de los Servicios Ofrecidos

En esta sección se especifica en detalle la configuración de los servicios que puede ofrecer un proveedor MPLS y se hace hincapié en los servicios implementados en la red (fig. 3-1) que utilizan los clientes para transmitir sus datos a sus sitios remotos.

4.3.4.4.1 Configurando y Habilitando VPNs en los PE – Routers

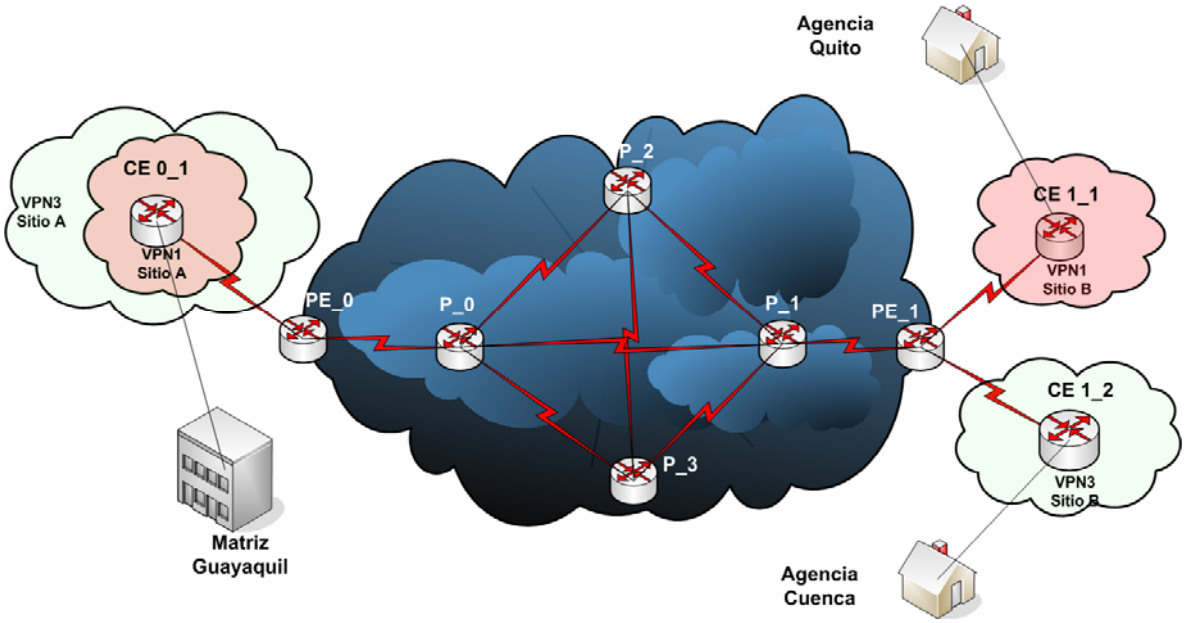


Figura 4-1.- Redes Privadas Virtuales (VPNs)

Para habilitar las redes privadas virtuales de capa de red, es necesario saber que, es entre los dispositivos de frontera que se realiza el intercambio de información de las intranets de los clientes conectados físicamente a aquellos LERs los cuales le harán parecer a los dispositivos CEs que están unidos a sus sitios remotos como si se tratara de un enlace dedicado.

Al tratar de entender el funcionamiento de las VPNs de capa de red se debe saber que cada una de ellas se asocia con una o mas instancias de enrutamiento y envío virtual, a la cual se las denomina VRF *(Virtual Routing and Forwading instances)*. Una VRF determina la membresía que tiene un cliente conectado a un ruteador de frontera del proveedor de servicio. Cada VRF está compuesta por una tabla de enrutamiento IP, una tabla CEF, un grupo de interfaces que utilizan dichas tablas, un conjunto de reglas y parámetros del protocolo de enrutamiento que controlan la información que se incluye en la tabla de ruteo. Las VRF contienen las rutas disponibles en la VPN que pueden ser accedidas por los sitios de los clientes. Cada sitio puede estar suscrito a varias VPN, pero solamente a una VRF (fig. 4-1). Para prevenir que no salga ni ingrese tráfico fuera de la VPN, cada VRF tiene guardada información de envío en las tablas IP y CEF.

La distribución de información de la conexión VPN de capa de red se controla mediante el uso de comunidades de ruta objetivo VPN. Las comunidades BGP extendidas se encargan de dicha distribución, mediante lo que se detalla a continuación:

- Cuando una nueva ruta VPN entra desde un ruteador CE, ésta ingresa al protocolo BGP y añade sus atributos a la lista de comunidades extendidas de ruta objetivo. Los valores de esta lista se obtienen de la lista de exportación de rutas objetivo relacionadas con la VRF de donde se obtuvo la nueva ruta.

- Adicionalmente, cada VRF incluye también una lista de importación de comunidades extendidas de ruta objetivo, la misma que define los atributos que una comunidad extendida de ruta objetivo debe tener para que la ruta pueda ser importada a la VRF.

Mediante una sesión entre el ruteador de frontera y el ruteador del cliente (PE – CE), el dispositivo ubicado en el borde del dominio MPLS obtiene el prefijo IPv4 del cliente para luego convertirlo en un nuevo prefijo VPN – IPv4 al añadirle 8 bytes de distintivo de Ruta (RD), que como su nombre lo indica, sirve para distinguir la ruta. Este nuevo prefijo sirve para identificar la dirección del cliente sin importar donde se encuentre y si su dirección es global o local, única o común. El RD se obtiene del VRF del ruteador PE en cuestión.

Para las VPNs en MPLS, BGP es el encargado de distribuir la información de capacidad de alcance a los prefijos VPN – IPv4. Cuando la distribución se lleva a cabo dentro del dominio IP – MPLS tenemos BGP interior por medio de sesiones entre los dispositivos de frontera (PE – PE),

Adicionalmente, mediante las extensiones multiprotocolo de BGP se lleva a cabo la propagación de la información alcanzable para proveer soporte a direcciones como IPv6 e IPX. Esta acción asegura que todos los miembros de la VPN reciban todas las rutas de las demás VPNs para que pueda haber comunicación entre todas.

Para el envío de paquetes en una conexión VPN con MPLS se hace uso de la información de ruteo almacenada en las tablas CEF y VRF. Los dispositivos de frontera añaden una etiqueta a cada prefijo que se obtiene de los ruteadores del cliente; el prefijo incluye información alcanzable de los demás ruteadores de frontera.

Los paquetes que atraviesan el *Backbone* MPLS llevan dos etiquetas, la primera tiene la dirección del ruteador de frontera que es el siguiente salto y la segunda que le indica cómo el ruteador PE alcanzado debe reenviar ese paquete al ruteador CE. Cuando el ruteador PE recibe el paquete etiquetado, lee la etiqueta, la quita y reenvía el paquete al destino marcado en la segunda etiqueta.

La creación y configuración de VPNs en MPLS es muy sencilla con el uso de BGP y se deben tener en cuenta pasos como: definición de VPNs, configuración de IBGP entre dispositivos de frontera, y configuración de enrutamiento hacia clientes en los ruteadores de frontera (tablas 4-12, 4-13, 4-14).

Definición de VPNs de capa de red	
Comando	**Propósito**
Router(config)# **ip vrf** <vrf-name>	Define la instancia de enrutamiento virtual con su nombre
Router(config-vrf)# **rd** <route-distinguisher>	Crea tablas de enrutamiento y envío
Router(config-vrf)# **route-target import** <route-target-ext-community>	Crea una lista de importacion de comunidades extendidas de ruta objetivo para la VRF especificada
Router(config-vrf)# **route-target export** <route-target-ext-community>	Crea una lista de exportación de comunidades extendidas de ruta objetivo para la VRF especificada
Router(config-vrf)# **interface** <type> <slot/port>	Ingresa al modo de configuración de interfaz
Router(config-if)# **ip vrf forwading** <vrf-name>	Asocia una VRF con una interfaz

Tabla 4-12.- Creación y definición de VPNs de capa 3

Configuración de MP - IBGP entre sesiones PE – PE	
Comando	**Propósito**
Router(config)# **router bgp** <*AS número*>	Ingresa al proceso de enrutamiento IBGP con el número de sistema autónomo que está configurado
Router(config-router)# **address-family vpnv4**	Ingresa al modo para configuración de MP - IBGP para VPNv4
Router(config-router-af)# **neighbor** <ip-address | peer-group-name> **activate**	Establece el emparejamiento con un vecino especificado.
Router(config-router-af)# **neighbor** <ip-address | peer-group-name> **send-community both**	Los vecinos renegocian sus capacidades

Tabla 4-13. – Configuración de Multiprotocol BGP

Configuración para la distribución de rutas hacia los clientes	
Comando	**Propósito**
Router(config)# **ip route vrf** <vrf-name> <destination-network> <destination-mask> <next-hop>	Define parámetros de ruta estática para cada sesión entre PE – CE
Router(config-router)# **address-family ipv4 unicast vrf** <vrf-name>	Define parámetros de ruta estática para cada sesión de enrutamiento BGP de PE a CE
Router(config-router-af)# **redistribute static**	Redistribuye las rutas estáticas de las VRF en la tabla BGP de la VRF
Router(config-router-af)# **redistribute connected**	Redistribuye las redes directamente conectadas en la tabla BGP de la VRF

Tabla 4-14.- Configuración de enrutamiento desde el Dominio MPLS hacia los sitios de los clientes

Cada una de las tablas anteriores da un detalle preciso de la configuración de las VPNs de capa de red en MPLS. Para su complemento, los *Anexos completos E, F y G* dan una referencia de las configuraciones, las pruebas de conectividad, así como también las rutas que se comparten mediante BGP.

4.3.4.4.2 Configurando y Habilitando Traffic Engineering.

La ingeniería de tráfico es esencial para los *backbones* de los proveedores de servicio porque son ellos los que deben soportar un elevado uso de las capacidades de transmisión, y las redes deben ser resistentes a las fallas que se den en los enlaces o en los nodos. Con la Ingeniería de tráfico MPLS las capacidades están integradas en la capa de red, lo cual optimiza el enrutamiento del tráfico dadas las obligaciones impuestas por la capacidad y la topología del *backbone*.

La ingeniería de tráfico MPLS direcciona el flujo del tráfico a través de una red basándose en los recursos disponibles en la misma y que el flujo de tráfico requiera, empleando enrutamiento basado en obligaciones en la cual, la ruta para un flujo de tráfico es la más corta que conoce y que se basa en requerimientos como: ancho de banda, prioridades; haciendo que los proveedores de servicio enrruten el tráfico de red, ofreciendo así lo mejor a los usuarios en términos de la tasa de transferencia y retardo, logrando ser mas eficientes y reduciendo costos de la red.

Por medio de la utilización del protocolo de reserva de recursos (RSVP), la ingeniería de tráfico MPLS establece y mantiene automáticamente LSPs a través del *Backbone*. El camino que un LSP utiliza (túnel) es determinado de acuerdo a los requerimientos y recursos de red, tal como el ancho de banda.

Los túneles de ingeniería de tráfico son calculados al inicio del LSP (cabecera de la ruta) basándose en un refuerzo entre recursos requeridos y disponibles. El protocolo de *gateway* interior encamina automáticamente el tráfico dentro de los túneles en donde un paquete cruzando una red MPLS

con ingeniería de tráfico viaja en un solo túnel que conecta el punto de ingreso con el punto de egreso.

4.3.4.4.2.1 Túneles de Ingeniería de tráfico

Para emplear la rapidez de la conmutación de etiquetas en lugar del encapsulamiento de capa de red, se puede implementar un túnel como una ruta conmutada de etiquetas (LSP) y hacer que el paquete viaje a través del mismo. El conjunto de paquetes que van a ser enviados por el túnel LSP corresponden a una FEC, para el cual, cada LSR que forma parte de la ruta debe asignar una etiqueta a esa clase (fig. 4-2).

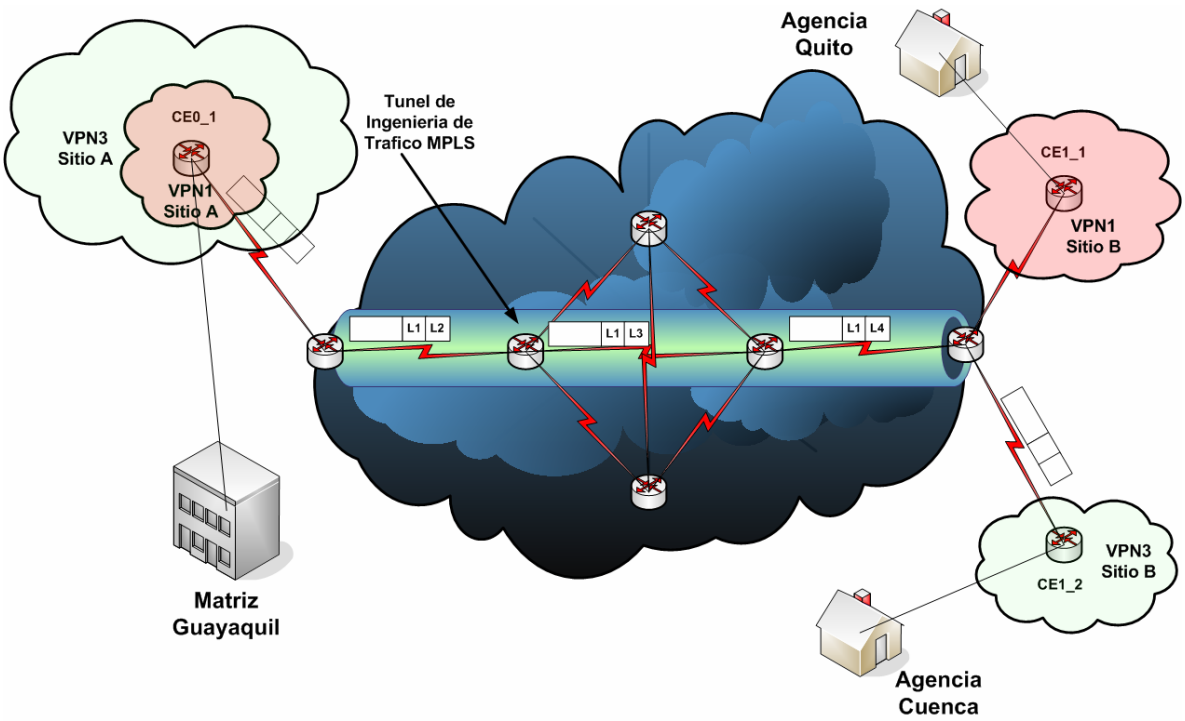


Figura 4-2.- Túnel de Ingeniería de Tráfico MPLS

4.3.4.4.2.1.1 Configuración de túneles de Ingeniería de Tráfico

Al configurar un dispositivo para soportar túneles de ingeniería de tráfico se deben desarrollar tareas sistemáticas que permitan entender el uso de estas herramientas. La tabla 4-15 muestra y detalla los pasos a seguir para habilitar túneles de ingeniería de tráfico en una red MPLS que utiliza al protocolo OSPF como protocolo de interiores.

Comando	Propósito
Router(config)# **ip cef**	Habilita de manera global una funcionalidad de envío y conmutación propietaria de Cisco (Requerida)
Router(config)# **mpls traffic-eng tunnels**	Habilita la herramienta para túneles de ingeniería de tráfico MPLS en un dispositivo.
Router(config)# **interface** <type> <slot/port>	Ingresa al modo de configuración de interfaz
Router(config-if)# **mpls traffic-eng tunnels**	Habilita los túneles de ingeniería de tráfico MPLS en una interfaz
Router(config-if)# **exit**	Regresa al modo de configuración global
Router(config)# **router ospf** <process-id>	Ingresa al modo de enrutamiento OSPF configurado
Router(config-router)# **mpls traffic-eng router-id loopback** <number>	Especifica que el identificador para el nodo es la dirección IP asociada con la interfaz loopback 0
Router(config-router)# **mpls traffic-eng area** <number>	Enciende el modo de ingeniería de trafico MPLS para un área de enrutamiento OSPF
Router(config-router)# **exit**	Regresa al modo de configuración global
Router(config)# **interface** <tunnel> <number>	Configura una interfaz túnel
Router(config-if)# **ip unnumbered loopback 0**	Asigna una dirección IP a la interfaz túnel
Router(config-if)# **tunnel destination** <ip address>	Especifica el destino para el túnel
Router(config-if)# **tunnel mode mpls traffic-eng**	Configura el modo de encapsulación del túnel a Ingeniería de Tráfico MPLS
Router(config-if)# **tunnel mpls traffic-eng bandwitdth** <bandwidth>	Configura el Ancho de Banda para el túnel de ingeniería de tráfico
Router(config-if)# **tunnel mpls traffic-eng priority** <number> <number>	Da prioridad a un túnel de ingeniería de tráfico MPLS
Router(config-if)# **tunnel mpls traffic-eng path-option** <number> **{dynamic | explicit}**	Configura el túnel para utilizar una ruta explícita o dinámicamente calculada desde la base de datos topológica de la Ingeniería de Tráfico
Router(config-if)# **tunnel mpls traffic-eng autoroute announce**	Especifica al IGP que utilice el túnel para reforzar el cálculo de la ruta mas corta

Tabla 4-15.- Configuración de Túneles de Ingeniería de Tráfico MPLS

4.3.4.4.2.1.2 RSVP

El uso del protocolo de reservación de recursos RSVP opera en cada salto de un túnel LSP, es utilizado para señalizar y mantener dichos caminos conmutados de etiquetas basándose en la ruta calculada. Para su configuración es necesario que se habilite las herramientas correspondientes a los túneles de ingeniería de tráfico en una Red MPLS. La tabla 4-16, describe el comando de configuración utilizado para habilitar el protocolo de señalización RSVP en una interfaz con funcionalidades de envío MPLS.

Comando	Propósito
Router(config)# **interface** <type> <slot/port>	Ingresa al modo de configuración de interfaz
Router(config-if)# **mpls traffic-eng tunnels**	Habilita los túneles de ingeniería de tráfico MPLS en una interfaz
Router(config-if)# **ip rsvp bandwidth** <bandwidth>	Habilita RSVP para una interfaz y especifica la cantidad de ancho de banda que será reservada

Tabla 4-16.- Configuración de Señalización RSVP

La interesante actividad de los túneles de ingeniería de tráfico en MPLS empleados para la señalización, pueden ser observados en el *Anexo H,* en el mismo que además de indicar la actividad de los túneles se especifica la ruta señalizada por el protocolo RSVP.

4.3.4.4.3 Configurando Calidad de Servicio (Quality of Service) (QoS).

La Calidad de Servicio (QoS) va ligada al tratamiento del tráfico en las redes, lo cual se lo hace en términos de Ancho de Banda, Latencia, Prioridad, Intermitencia, Perdida de Paquetes, entre otros. Para la implementación de

QoS existen tres pasos fundamentales que permiten tratar eficientemente el flujo de tráfico en una red y los mismos se mencionan a continuación:

- **Clasificación:** Identificar y Marcar los paquetes para lo cual, variados niveles de servicios pueden ser reforzados a través de una red.

- **Organización:** Asignar paquetes a alguna de múltiples colas y tipos de servicios asociados, basándose en la clasificación para un tratamiento con nivel de servicio específico en la red.

- **Gestión de Recursos:** Esmeradamente calcular el ancho de banda requerido para todas las aplicaciones.

La herramienta de Calidad de Servicio utiliza varios componentes para seguir metódicamente los pasos mencionados anteriormente. Dichos componentes como: Clasificación del Tráfico, Gestión de la Congestión, Formación y Organización del Tráfico, y Prevención de la Congestión ayudan a evitar una saturación en la red, distribuyendo el tráfico según las políticas dictadas para la red.

4.3.4.4.3.1 Clasificación y Marcación del tráfico (Classification and Marking).

Muchas de las herramientas de QoS que permiten clasificar el tráfico, permiten a cada clase de tráfico recibir un diferente nivel de tratamiento. Estos diferentes tipos o clases de tráfico son típicamente llamados “Clases de Servicio”. La clasificación permite que los dispositivos de redes decidan quienes son los paquetes pertenecientes a cada parte de cada clase de servicio.

Las herramientas de clasificación y marcación, no solamente clasifican paquetes en clases de servicios, sino que además marcan los paquetes que están en una misma clase de servicio con el mismo valor en un campo en la cabecera del paquete. La marcación de paquetes hace que otras herramientas de QoS puedan examinar los *bits* marcados del paquete para clasificar el tráfico en una forma más fácil.

La mayoría de las herramientas de clasificación en QoS, utilizan algún grado, dado que, para colocar paquetes en distintas colas el sistema operativo en los dispositivos de red (IOS) debe diferenciar entre paquetes. La lógica utilizada cuando los datos ingresan a una red encapsulados puede ser descrita de la siguiente manera:

- Para paquetes que ingresan por una interfaz, si ellos coinciden con un criterio de emparejamiento de alguna clase, ellos son marcados con algún valor.

- Si en primera instancia el paquete no coincide con el criterio de emparejamiento, se sigue comparando con las demás clases hasta que sea emparejado.

- En caso de que el paquete no cumpla con ningún criterio, la acción que es tomada será enviarlo como si no existiera alguna herramienta de QoS configurada.

La clasificación y marcación basada en clases de servicio examina y distribuye las clases al tráfico, chequeando las cabeceras de los paquetes. La tabla 4-17 enlista los campos que pueden ser tomados en cuenta al momento de clasificar el tráfico de paquetes, y los *Anexo J-1-3, J-2-4, J-3-1,*

J-3-2, muestran el viaje de lo paquetes marcados a través del núcleo de la red.

Campo	Comentario
IP DSCP	Campo del paquete IP utilizado para QoS
IP Precedence	Campo del paquete IP utilizado para QoS
MPLS EXP	Campo de la Cabecera MPLS utilizado para QoS
NBAR Protocol Types	Herramienta utilizada para reconocimiento de protocolos

Tabla 4-17.- Campos de Calidad de Servicio usados para la clasificación y marcación del tráfico

4.3.4.4.3.1.1 Network Based Application Recognition (NBAR)

La herramienta de clasificación y marcación basada en clases de servicio puede ser configurada en los campos mostrados en la tabla 4-17, para directamente clasificar los paquetes. Sin embargo, el Marcado basado en clase (CB Marking) puede también utilizar un reconocimiento de las aplicaciones ejecutándose en una red o NBAR *(Network Based Application Recognition)* para clasificar paquetes. Independientemente, NBAR puede ser configurado para mantener contadores de tipo de tráfico y volumen de tráfico para cada tipo.

NBAR es una herramienta de clasificación, que reconoce y clasifica una amplia variedad de protocolos y aplicaciones, incluyendo aplicaciones basadas en *Web* y otras difíciles de clasificar y protocolos que usan asignaciones de puertos dinámicos TCP y UDP. Una vez que NBAR reconoce y clasifica un protocolo o aplicación, la red puede ser configurada

para aplicar la apropiada herramienta de Calidad de Servicio para aquella aplicación o tráfico con cualquier protocolo. Por ejemplo, algunas aplicaciones usan números de puertos dinámicos, así un comando ***match*** configurado estáticamente, el cual busca un número particular de puerto UDP o TCP, simplemente no podría clasificar el tráfico. NBAR puede chequear anticipadamente la cabecera UDP y TCP realizando una inspección profundizada del paquete y reconoce además la información específica de una aplicación.

La calidad de servicio es aplicada utilizando la herramienta de "Interfaz de Línea de Comandos Modular de QoS (MQC)", que es ventajosa ya que usa métodos de clasificación y marcación basándose en la clase *(CB-Marking).* Es fácil reconocer cuando *CB Marking* está utilizando clasificación por medio de NBAR. Cuando el comando ***"match protocol"*** es utilizado, *CB Marking* esta buscando emparejar un protocolo descubierto por NBAR.

Para la implementación de las herramientas de Clasificación y Marcación, los comandos referidos en las tablas 4-18, 4-19, indican la forma más fácil de clasificar los paquetes con la utilización de NBAR.

Mapa de Clases	
Comando	**Proposito**
Router(config)# **class-map** <class-map-name>	Crea un mapa de clase para tráfico
Router(config-cmap)# **match protocol** <protocol-name>	Especifica el uso de NBAR para reconocer el protocolo de aplicación

Tabla 4-18.- Creación de Mapa de Clases

Mapa de políticas a la entrada	
Comando	**Proposito**
Router(config)# **policy-map** <policy-map-name>	Crea un mapa de políticas para el tráfico de entrada
Router(config-pmap)# **class** <class-map-name>	Asocia la política de entrada que se le asignara a cada clase
Router(config-pmap-c)# **set ip precedence** <priority-mark>	Asigna la Marcación al paquete entrante con un valor de prioridad según la clase

Tabla 4-19.- Creación de Mapa de Políticas

4.3.4.4.3.2 Gestión de la Congestión (Congestion Management).

Entre las herramientas de mayor importancia de QoS esta la “Gestión de la Congestión”, cuyo término se usa para referirse a un sistema de encolamiento. Esta herramienta permite que se controle la congestión determinando el orden en el que los paquetes son enviados fuera de una interfaz, basándose en prioridades asignadas a cada paquete. La administración del congestionamiento entabla la creación de colas, la asignación de paquetes a dichas colas basándose en la clasificación de los mismos, y la organización de dichos paquetes en la cola para la transmisión; para lo cual la herramienta ofrece 4 tipos de protocolos de encolamiento, cada uno de los cuales especifica la creación de diferentes números de colas que se esfuerzan por diferenciar y especificar el orden en el cual el tráfico es enviado.

El por qué debe usarse mecanismos de encolamiento radica en que las redes hoy en día pueden incluir algunos protocolos utilizados por aplicaciones, dando crecimiento a la necesidad de priorizar el tráfico para

satisfacer a las aplicaciones de misión crítica, mientras se está direccionando las necesidades de las aplicaciones que dependen de menos tiempo (ej. FTP). Los diferentes tipos de tráfico compartiendo una ruta de datos a través de la red pueden interactuar con otros de una forma que no afecte al desempeño de sus aplicaciones. Si una red es diseñada para soportar diferentes tipos de tráfico, los mismos que comparten una sola ruta entre ruteadores, se debe considerar el uso de técnicas de Gestión de Congestión para asegurar la igualdad en el trato de varios tipos de tráfico.

Existen amplios factores que se consideran al momento que se desee implementar la herramienta de calidad de servicio en mención *(Congestion Management):*

- La priorización del tráfico es mayormente efectiva en enlaces seriales (WAN), donde la combinación del tráfico en grandes proporciones y las tasas de transmisión relativamente bajas pueden temporalmente ser causa de congestión.

- Depender del promedio del tamaño del paquete, la priorización es mas efectiva cuando se aplica a enlaces con anchos de banda a niveles de E1 / T1 o menores.

- Si los usuarios de aplicaciones ejecutándose a través de la red notan un tiempo de respuesta bien pobre, se debería considerar el uso de herramientas para gestionar la congestión. Estas herramientas son dinámicas, y se adaptan a las condiciones existentes en la red. Sin embargo, hay que considerar que si el enlace WAN permanece constantemente congestionado, la priorización del tráfico no podría resolver el problema. El aumento de ancho de banda podría ser la solución apropiada.

- Si la congestión en el enlace serial (WAN) no existe, no hay ninguna razón para implementar priorización del tráfico.

Así como existen factores que deben ser considerados al momento de implementar herramientas para gestionar la congestión, existen además aspectos relevantes que deberían considerarse en la determinación de si deber ser o no establecida e implementada una política de encolamiento para la red.

- Determinar si la WAN está congestionada, situación que se presenta cuando los usuarios perciben una degradación en el desempeño de ciertas aplicaciones.

- Determinar los objetivos y metas basándose en la mezcla del tráfico que se necesita gestionar, la topología y diseño de red, para lo cual se debe:

    - Establecer distribución equitativa de asignación de ancho de banda a través de todos los tipos de tráfico que se identifique.

    - Garantizar estricta prioridad al tráfico de especial tipo de aplicaciones.

    - Personalizar la asignación del ancho de banda, para lo cual los recursos de la red son compartidos entre todas las aplicaciones a las que se le da servicio.

    - Configurar efectivamente el encolamiento. Se debe analizar los tipos de tráfico utilizando la interfaz y determinar como distinguirlo. Para ello se debe haber aplicado mecanismos de clasificación primeramente.

- Configurar la interfaz para que utilice el tipo de estrategia de encolamiento que se ha escogido, y observar los resultados.

Durante periodos de bajo tráfico, donde no existe congestionamiento, los paquetes son enviados fuera de una interfaz tan pronto como ellos arriban a la misma. Durante periodos de congestionamiento en la transmisión por la interfaz saliente, los paquetes arriban más rápido de lo que la interfaz los puede enviar. Si las herramientas de gestión de la congestión son utilizadas, los paquetes que se acumulan en una interfaz son encolados hasta que las misma es liberada para enviarlos; ellos son organizados para la transmisión de acuerdo a su asignación de prioridad y al mecanismo de encolamiento configurado para la interfaz. Un ruteador determina el orden de la transmisión de paquetes controlando, cómo deben ser localizados en una cola, y cómo las mismas deben ser servidas respetando a las demás. Como se presenta tempranamente, la “Gestión de la Congestión” consiste en cuatro mecanismos de encolamiento los mismos que se detallan a continuación:

***FIFO (First In First Out)* – (Primero en entrar, primero en salir):** Este mecanismo no entabla ningún concepto de prioridad o clases de tráfico. Con FIFO, la transmisión de paquetes fuera de una Interfaz ocurre en el orden en que los mismos arriban.

***WFQ (Weighted Fair Queueing)*:** WFQ ofrece encolamiento dinámico y equitativo, para lo cual divide el ancho de banda para todas las colas basándose en pesos. Para comprender cómo trabaja WFQ, se considera la cola para una serie de paquetes FTP como una cola colectiva, y para el tráfico interactivo discreto de paquetes como una cola individual. Dados los pesos de las colas, WFQ asegura que para todos los paquetes FTP

enviados colectivamente, un número igual de tráfico de paquetes interactivos individuales es enviado.

Dada esta manipulación, WFQ asegura satisfactorios tiempos de respuesta a las aplicaciones críticas, tales como: las interactivas, las aplicaciones basadas en transacciones, las cuales no son tolerantes de degradación en su desempeño. Para interfaces seriales a nivel de E1 (2.048 Mbps) y menores, WFQ basado en flujos es utilizado por defecto. Cuando ninguna otra estrategia de encolamiento es utilizada, todas las demás interfaces utilizan FIFO por defecto. Para la organización WFQ es subdividido en cuatro tipos:

- WFQ basado en flujos

- WFQ distribuido (DWFQ)

- WFQ basado en clases (CBWFQ)

- WFQ distribuido basado en clases (DCBWFQ)

***CQ (Custom Queueing)*:** Con este mecanismo de encolamiento, el ancho de banda es asignado proporcionalmente para cada clase de tráfico. CQ permite que se especifique el número de *bytes* o paquetes que son trazados desde la cola, lo cual es especialmente muy usual en interfaces lentas.

PQ (Priority Queueing): El encolamiento de prioridad PQ, permite que los paquetes pertenecientes a una clase de tráfico con prioridad sean enviados antes de los paquetes que tienen una menor prioridad asegurando así la entrega a tiempo de aquellos paquetes.

Una vez presentados los cuatro mecanismos de encolamiento es necesario saber cual es el método más efectivo que será utilizado dependiendo de la aplicación que se tenga. Es necesario un resumen comparativo de las técnicas de encolamiento que ayude a decidir el mecanismo que será de gran utilidad. En el encolamiento FIFO, las fuentes pueden consumir todo el ancho de banda disponible, las ráfagas que producen las fuentes causan retardo en el tráfico de importancia, y éste puede ser descartado dado que el tráfico de menor importancia llena la cola.

Para decidir si se utilizará CQ, PQ o WFQ debe atenderse los siguientes aspectos:

- CQ garantiza algunos niveles de servicio a todo el tráfico ya que se puede asignar ancho de banda a todas las clases de tráfico. Se puede definir el tamaño de la cola, determinando su capacidad de conteo de paquetes en lugar de controlar el acceso al ancho de banda.

- PQ garantiza prioridad estricta, en la cual asegura que un tipo de tráfico será enviado, posiblemente a expensas de los demás. Para PQ una cola de prioridad baja puede verse muy afectada, y en el caso, nunca será permitida a enviar sus paquetes si una cantidad limitada de ancho de banda está disponible o si la tasa de transmisión del tráfico crítico es alta.

- WFQ no requiere configuración de filtros de tráfico (listas de acceso) para determinar lo preferido en una interfaz serial. En lugar de ello, el algoritmo equitativo de la cola, dinámicamente clasifica el tráfico en mensajes que son parte de una conversación. El bajo volumen de tráfico obtiene asignación equitativa de ancho de banda con WFQ, como lo hace el tráfico de alto volumen, como la transferencia de archivos.

La siguiente tabla compara los mecanismos mencionados anteriormente.

Mecanismos de Encolamiento	WFQ basado en Flujo	CBWFW	CQ	PQ
Número de Colas	Tiene un configurable número de colas de 256	Puede configurarse una cola por clases, hasta 64 clases	16 Colas de Usuario	4 Colas
Tipo de Servicio	Asegura igualdad entre todo el tráfico basándose en pesos. La prioridad estricta del encolamiento está disponible a través del uso de características de prioridad tales como IP RTP	Proporciona garantía de ancho de banda para las clases de tráfico definidas por el usuario Proporciona soporte WFQ para clases de tráfico no definidas por usuarios. El encolamiento estricto de prioridad está disponible a través de características como IP RTP y LLQ	Servicio *Round Robin*	Colas de Alta prioridad. Son servidas primero.
Configuración	No requerida	Requerida	Requerida	Requerida

Tabla 4-20.- Comparación entre mecanismos de encolamiento

Para propósitos de gestionar la congestión, en caso de que ocurra; en la red MPLS (fig. 3-1) se aplican herramientas que permitan diferenciar el tráfico en clases diferentes de servicio, para lo cual se utiliza CBWFQ, conjuntamente con PQ (LLQ), dado que permite trabajar clasificando y dando prioridad a aplicaciones de misión critica y que requieren baja latencia como por ejemplo la Voz y el Video.

La implementación de las herramientas de Gestionamiento de la congestión es muy sencilla e involucra: la creación de mapas de clases, la implementación de una política de clases en el mapa de políticas y la adición de la política de servicio conjuntamente con la habilitación de la herramienta en la interfaz de salida de los paquetes. La tabla 4-21 muestra los comandos utilizados en los equipos Cisco 7206VXR para la configuración de esta herramienta de Calidad de Servicio.

Comando	Propósito
Router(config)# **class-map** <class-map name>	Especifica el nombre del mapa de clases
Router(config-cmap)# **match mpls experimental** <number>	Expecifica el valor del campo EXP que será utilizado como criterio de igualdad para determinar si los paquetes chequeados pertenecen a una clase específica
Router(config)# **policy-map** <Policy-map-name>	Especifica el nombre del mapa de políticas
Router(config-pmap)# **class** class-name	Especifica el nombre de la clase predefinida incluida en la política de servicio
Router(config-pmap-c)# **bandwidth** {<bandwidth-kbps> | **percent** <percent>}	Especifica la cantidad de ancho de banda o porcentaje disponible del mismo que será asignado a una clase.
Router(config-pmap-c)# **priority** {<bandwidth-kbps> | **percent** <percent>}	Crea una clase de prioridad estricta y especifica la cantidad de ancho de banda o porcentaje que será asignado a una clase

Tabla 4-21.- Configuración de Herramientas de Encolamiento y Gestión del Tráfico

Al momento de la configuración puede utilizarse los comandos *"bandwidth"* para especificar que se trabajara con CBWFQ y *"priority"* para indicar que se utilizara una cola de prioridad LLQ. La diferencia radica en que CBWFQ puede tomar ancho de banda disponible de las clases que no estén utilizando recursos para las aplicaciones que no se estén ejecutando. Por otro lado LLQ es utilizado para aplicaciones de alta prioridad como lo son la voz y el video, en lo cual se especifica una cantidad de ancho de banda y esta técnica no permitirá que las aplicaciones de misión crítica se excedan en el uso de los recursos de la red.

4.3.4.4.3.3 Organización y Creación de Tráfico (Traffic Policing and Shaping)

Para la implementación de QoS siempre es necesario contar con herramientas que ayuden a que se mantengan y se cumplan los contratos de tráfico en una red. Herramientas de Organización, Creación y Formación *(Policing, Shaping),* permiten que los dispositivos pertenecientes a una red se ajusten a un contrato estipulado de tráfico de información. El contrato de tráfico define la cantidad de información (datos) que pueden ser enviados desde una red a otra; típicamente se expresa como la tasa de información obligada o fija (CIR) con una ráfaga obligada (Bc). De las técnicas de *Policing,* se conoce que las mismas miden el flujo de datos, y descartan los paquetes que exceden el contrato de tráfico, mientras que las técnicas *Shaping* permiten que los paquetes se ajusten a un contrato de tráfico. En ciertos casos, en los que los paquetes que excedan un contrato de tráfico puedan ser descartados, el dispositivo de envío puede optar por disminuir su tasa de transmisión, para lo cual los paquetes no serán desechados.

En términos más sencillos de interpretación, la técnica conocida como *Policing* descarta paquetes que exceden un contrato de tráfico *(Dropping)*,

contrariadamente a lo que hace la técnica de *Shaping* que retarda el envío de información disminuyendo la prioridad y ubicando en *Buffers* los paquetes excedidos *(Queueing)*. En ambos casos las herramientas de QoS que regulan y forman el tráfico previenen que se exceda la tasa de transmisión estipulada.

Tanto los reguladores o organizadores y los formadores o creadores de tráfico *(Policers, Shapers)* son muy aplicables en las fronteras entre dos redes diferentes. Su método de implementación y configuración requiere de un ligero entendimiento de una técnica muy usual que se emplea para que los dispositivos de redes se ajusten a un CIR. La técnica de "Balde de Peticiones de Envío *(Token Bucket)*" que utilizan los reguladores y formadores del tráfico, es una definición formal de una tasa de transferencia, la cual consta de tres componentes principales: Un tamaño de ráfaga (Bc), una tasa de transferencia media (CIR), y un intervalo de tiempo para envío (Tc). Aunque la tasa de transferencia media generalmente es representada en *bps*, cualquiera de los dos valores restantes puede ser derivado del tercero como se indica en la formula:

$$mean\_rate = \frac{burst\_size}{time\_\_\text{int}erval}$$

Es necesario llevar presente que por definición, la tasa de bits de la interfaz no excederá a la tasa media, y la misma puede acoplarse arbitrariamente dentro del intervalo de transmisión de ráfaga.

La técnica *Token Bucket* es utilizada para gestionar un dispositivo que regula los datos en un flujo. La técnica por si misma no descarta ni organiza políticas, en lugar de ello descarta peticiones de envío *(Tokens)* y le deja al flujo de información el problema de gestionar su cola de transmisión en caso de que el mismo sobrepasa el regulador *(Policer)*.

Para comprender mejor la técnica hay que pensar que las peticiones de envío son colocadas en un balde a una cierta tasa. El balde tiene una capacidad específica, la misma que si es llenada al máximo de su capacidad no permitirá que ingresen mas peticiones de envío *(Tokens)*, descartándolas. Dado que cada petición es autorizada por una fuente para que envíe un cierto número de bits por la red, al momento de enviar un paquete se debe remover del balde, un número igual de *Tokens* en representación al tamaño del paquete. En caso de que no exista una gran cantidad de peticiones en el balde, el paquete esperará hasta que se tenga una gran cantidad de *Tokens*, y al encontrarse completamente lleno el balde, los *Tokens* entrantes no estarán disponibles ni siquiera para futuros paquetes. En cualquier momento, la ráfaga más grande que una fuente pueda enviar en la red se acoplará completamente al tamaño del balde.

La diferencia entre las dos herramientas *(Traffic Policing y Traffic Shaping)* radica en la utilización de Colas o Buffer de Datos conjuntamente trabajando con la técnica *Token Bucket* en lo referente a formación (*Shaping)* y de esa manera los paquetes que arriban y no pueden ser enviados inmediatamente serán retardados en el Buffer. Para lo concerniente a *Traffic Shaping,* la técnica de balde de peticiones de envío permite enviar ráfagas pero limitando estas, lo cual garantiza que las mismas están limitadas para que nunca se envíe mas rápido de los que permite la técnica implementada, entre el intervalo de tiempo, mas la tasa establecida a la cual las peticiones son ubicada en el balde.

$$\left(token\_bucket\_capacidad(bits) \Big/ \text{int}ervalo\_tiempo(s) \right) + tasa\_establecida(bps) = máxima\_velocidad\_flujo(bps)$$

Este método de limitar las ráfagas también garantiza que la tasa de transmisión en términos de longitud no exceda a la tasa establecida en la que las peticiones son localizadas en el balde.

Después de comprender la técnica que utilizan los reguladores y modeladores del tráfico se puede proceder con la creación de las políticas de tráfico, las mismas que deben ser añadidas a una interfaz específica. Para el tráfico que ingresa a la red MPLS (fig. 3-1) las políticas de tráfico que permitirán regular el mismo han sido configuradas en el mapa de políticas de entrada de los ruteadores de frontera, de la siguiente forma:

Comando	Propósito
Router(config-pmap-c)# **police** bps busrt-normal burst-max **conform-action** action **exceed-action** action **violate-action** action	Especifica una utilización máxima del ancho de banda para una clase de tráfico. Utiliza la técnica Token Bucket, cuyas variables son seteadas en la linea de comandos.

Tabla 4-22.- Configuración de Herramientas de Organización del tráfico

La tabla 4-22 muestra el comando utilizado en el mapa de políticas de un ruteador para la configuración de la herramienta de QoS denominada *traffic Policing.*

La implementación de modeladores, creadores o formadores del tráfico se pueden realizar en las interfaces de salida de los dispositivos que se conectan a una red (podría ser en los dispositivos CEs) y sus modos de implementación se resume en la tabla 4-23:

Comando	Propósito
Router(config-pmap-c)# **shape {average | peak}** cir [bc] [be]	Especifica el promedio o pico de la tasa de modelamiento o formación de tráfico (Shaping)

Tabla 4-23.- Configuración de Herramientas de Modelamiento y formación del tráfico

4.3.4.4.3.4 Prevención de la Congestión (Congestion Avoidance)

Para prevenir la congestión las herramientas de calidad de servicio se valen de técnicas que ayudan a las colas a gestionar el tráfico y a prevenir el congestionamiento. Las colas se llenan cuando la carga cumulativa de los emisores de paquetes, exceden la tasa límite de una interfaz. Cuando la mayor parte del tráfico necesita salir de una interfaz a una tasa de transmisión superior a la soportada por la misma, ocurre la formación de las colas. Las técnicas de encolamiento ayudan a gestionar el tráfico en las colas, mientras que las técnicas de prevención ayudan a reducir el nivel de congestión en las colas, selectivamente descartando paquetes.

Las técnicas de prevención de congestión en QoS, dan la oportunidad de realizar negociaciones entre características como la pérdida de paquetes versus el retardo e intermitencia, sin embargo, la negociación no es simple ya que se tiene que descartar paquetes antes de llenar completamente la cola para lo cual la pérdida de paquetes cumulativa puede ser reducida completamente con el retardo e intermitencia. Las técnicas de congestión es cierto que descartan paquetes pero a la vez ayudan a alcanzar el efecto de tener una red saludable. Entre los tipos de mecanismos conocidos para prevenir la congestión se presentan brevemente los siguientes.

Tail Drop: Es un mecanismo utilizado por defecto en los dispositivos de red.

WRED: Combina las capacidades de los algoritmos de detección temprana con los valores de prioridad que se localizan en el campo IP Precedence localizado en la parte de tipo de servicio ToS de la cabecera IP. De los algoritmos de detección temprana el más utilizado es el “WRED basado en flujos” ya que el mismo proporciona gran igualdad a todos los flujos en una interfaz sin importar la cantidad de paquetes que sean descartados.

WRED es muy usual en interfaces de tráfico saliente, donde se espera que ocurra el congestionamiento de la información como lo es el núcleo de la red. Dado que los dispositivos de frontera asignan las prioridades al clasificar y marcar los paquetes, los dispositivos del núcleo de red con la técnica de descarte de paquetes implementada, determinan la forma en la cual deben tratar a cada tipo de tráfico. En la prevención de congestión, WRED cuenta con umbrales separados y pesos para diferentes prioridades, permitiendo que se ofrezcan diferentes calidades de servicio considerando el desecho de paquetes para los diferentes tipos de tráfico. El tráfico estándar puede ser descartado más frecuentemente que el tráfico de alta prioridad durante los periodos de congestión.

La manera de desempeño y trabajo de las técnicas de detección temprana, se basa en el descarte aleatorio de paquetes antes de que ocurran los periodos de alta congestión advirtiendo a las fuentes para que disminuyan su tasa de transmisión *(Traffic Shaping)*. Si el paquete fuente utiliza TCP como protocolo de transporte, se disminuye la transmisión hasta que todos los demás paquetes alcancen su destino, indicando la aclaración o limpieza de la congestión. Generalmente los paquetes son descartados selectivamente basándose en el contenido del campo *“IP Precedence”*. Así los paquetes con más alta prioridad, son más posibles a ser entregados satisfactoriamente que los paquetes que ingresan a una red con una menor prioridad. En la figura 4-3 se ilustra la forma de trabajo de WRED.

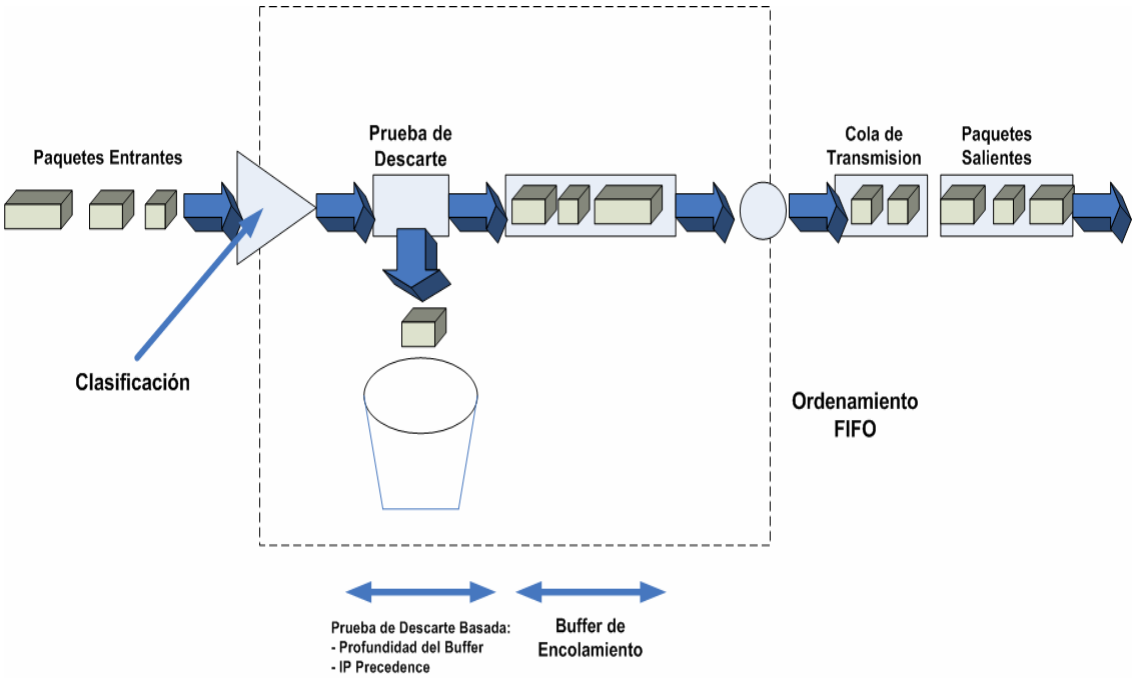


Figura 4-3. WRED

Cuando un paquete arriba a una interfaz, ocurren una serie de eventos:

- El promedio del tamaño de la cola es calculado.

- Si el promedio es menor que el umbral mínimo de la cola, los paquetes arribando son encolados.

- Si el promedio está entre el umbral mínimo de la cola y el umbral máximo para la interfaz, el paquete es descartado o encolado, dependiendo de la probabilidad de desecho del paquete.

- Si el promedio del tamaño de la cola es mayor que el umbral, el paquete es descartado.

En cuanto a configuración, WRED es tan sencillo de implementar y se resume en un comando que permite su habilitación (tabla 4-24).

Comando	Propósito
Router(config - if)# **random-detect**	Habilita WRED

Tabla 4-24. – Configuración de Herramientas de Prevención de Congestión

Las técnicas de calidad de servicio y las clases en las que se distribuye el tráfico que atraviesa la red MPLS, pueden ser analizadas en los mapas de políticas y de clases en el *Anexo I-1.*

4.3.4.5 Configuración de los Routers de los Clientes.

Para los sitios de los clientes que desean conectar sus empresas por medio de una red MPLS que les brinde seguridad al momento de transmitir la información es necesario conocer los diferentes dominios que se pueden presentar al momento de entablar las redes de datos. Para el diseño presentado (fig. 3-1, Pag. 94) se presentan el dominio IP de los sitios de los clientes y un dominio MPLS por medio del cual se enlazaran los puntos locales con los sitios remotos.

4.3.4.5.1 Configurando los CEs (Customer's Equipments)

La configuración de los dispositivos localizados en las oficinas de los clientes es muy sencilla y puede ser vista como un enrutamiento sencillo del tráfico IP hacia un destino final, es decir de un punto a otro. Para el cliente la información de etiquetamiento MPLS es transparente ya que los mismos no pertenecen al dominio en el cual se asignan y se conmutan las etiquetas a los paquetes que los envían hacia la red MPLS. La información de

enrutamiento de los dispositivos que se conectan a la nube MPLS es enviada de extremo a extremo dentro de la red mediante los protocolos de enrutamiento ya mencionados (BGP, OSPF).

4.3.4.5.1.1 Conectividad hacia la Red MPLS

La información desde los clientes puede ser enviada mediante protocolos de enrutamiento dinámico (BGP exterior, OSPF entre áreas, RIP, etc.), o mediante la conectividad estática como es este caso. Una vez que la información sea recibida por los ruteadores de frontera de la red, los mismos se encargaran de enrutar el tráfico hacia los equipos del núcleo MPLS con un trato específico acorde a la calidad de servicio que puede brindar el dominio.

4.3.4.5.1.1.1 Configurando el Enrutamiento

Dado que se ha decidido enrutar estáticamente el tráfico de la red de los clientes, el comando de configuración presentado en la tabla 4-25 ayuda a direccionar los paquetes hacia la red MPLS, para que luego el equipamiento dentro de la misma se encargue de direccionarlo a su destino.

Comando	Propósito
Router(config)# **ip route** <destination-network> <destination-mask> <next-hop>	Enruta los paquetes estáticamente siempre y cuando se especifique la red y máscara de destino y el siguiente salto

Tabla 4-25.- Configuración del Enrutamiento desde los clientes hacia la nube MPLS

Para tener la idea clara acerca de la conectividad hacia la red MPLS, puede referirse al *Anexo C* en el cual se detalla la configuración de un dispositivo ubicado en las instalaciones de los clientes

De las implementaciones de los servicios que puede brindar la red MPLS de un proveedor se pueden obtener muchos resultados. El hecho de que un cliente piense que sus empresas poseen enlaces dedicados para conectarse con sus agencias, es uno de los objetivos principales de la implementación de conexiones VPNs entre la matriz y las agencias (fig. 3-2). Las agencias de una empresa típica pueden acceder a los distintos aplicativos que su matriz les pueda ofrecer por medio de conexiones VPN en MPLS.

Para la validación de esta implementación y para la obtención de buenos resultados, se emulan aplicaciones de correo, transferencia de archivos, mensajería instantánea, etc., tal como se da en el típico esquema empresarial que se pueda tener hoy en día en donde, los usuarios de una agencia descargan sus correos desde un servidor principal ubicado en la matriz por medio de una conectividad VPN entre los dos puntos mencionados (agencia – matriz). De igual manera se emula el servicio de transferencia de archivos dado que existen muchas ocasiones en las que se necesite contar con el acceso a base de datos y archivos.

La información que es enviada a través de una red MPLS debe ser clasificada al ingresar a la nube para ser servida de la manera más eficiente según la prioridad (QoS) que el proveedor de servicios haya implementado en su dominio evitando de esa forma saturar su red proveedora de transporte. Como se puede notar en el *Anexo J-1* las aplicaciones de envío de correo son atendidas con una prioridad de nivel 3 ya que pueden existir aplicaciones de voz y video que deberían tener prioridad sobre las demás, de manera que no sufran ningún tipo de perdidas.

Otra de las aplicaciones que hoy en día es muy usual encontrar, es la administración remota, para lo cual se ha dado un nivel de prioridad un poco menor al envío de correos, siendo también de mucha importancia en el campo de la gestión en estos días. La administración remota ha sido emulada con la aplicación *"TELNET"* a la cual se le asigna cierto porcentaje de ancho de banda en la red de un proveedor para que no afecte a aplicaciones que puedan demandar una mayor velocidad en la transmisión de información.

En los *Anexos J-1* se detalla el servidor SMTP y POP3 utilizado para emular la aplicación de correos electrónicos, lo cual es muy usual en las empresas. Además con el analizador de tráfico *Wireshark* se puede hacer el análisis del tráfico en la interfaz de un ruteador dentro de la nube MPLS notando de esa manera que un paquete de correo lleva la etiqueta MPLS correspondiente a su FEC y además puede observarse que el paquete lleva en el campo de calidad de servicio de MPLS (EXP) su respectiva marca de prioridad asignada en el ingreso al dominio.

Se da a conocer en la emulación de bases de datos de archivos, el servidor FTP utilizado para estos propósitos, para lo cual también se puede notar la marca de calidad de servicio con el analizador de tráfico antes mencionado. Véase el *Anexo J-2.*

La administración remota de las agencias de una empresa también es analizada con *Wireshark,* y, al igual que las aplicaciones de correos, transferencia de archivos, voz y video; se puede notar que tiene una prioridad asignada según su correspondiente calidad de servicio y el trato que le da la red del proveedor. Véase el *Anexo J-3.*

Nota: Pueden probarse un mayor número de aplicaciones en mundo real. Con dynamips las pruebas de Calidad de Servicio se limitan a la observación de los paquetes MPLS marcados en su campo EXP dado que el procesamiento del simulador depende de los recursos de la PC en donde se ejecuta.

4.4 Resumen de la Implementación del Diseño de la Red mediante Dynamips.

La validación de un diseño de red MPLS puede ser realizado con herramientas simulación muy poderosas que garanticen confianza a la hora de emular el diseño. Herramientas como *Dynamips / Dynagen*, las cuales son configurables, rigurosas, portables, etc., permiten analizar los datos de una red tal y como se trabajara en un entorno real con equipos de plataforma Cisco. El poder de estas herramientas radica en el uso directo del sistema operativo (IOS) de los dispositivos de enrutamiento, logrando además que se pueda simular las conexiones del equipamiento y se haga un emparejamiento *(matching)* de entornos real y virtual al momento de la emulación.

A la hora de simular el diseño con *Dynamips / Dynagen* se puede pensar que se trabaja en un ambiente real, y las tareas que se realizan en entornos reales pueden ser simuladas también garantizando que el funcionamiento de una red tenga un desempeño adecuado al momento de tener una implementación con dispositivos reales. Tareas como la habilitación de protocolos de enrutamiento OSPF, BGP; protocolos de señalización y distribución de etiquetas como LDP y RSVP, pueden ser implementados, ayudando de esa manera a tener ambientes que además podrían incurrir en el uso de aplicaciones como redes privadas virtuales (VPN), Ingeniería de Tráfico, y Calidad de Servicio para información que pueda atravesar la red MPLS al momento de realizar simulaciones de ruteadores de clientes que sean capaces de generar tráfico.

La capacidad que hace que el simulador sea analizable, es de gran ayuda ya que refuerza los conceptos en lo referente a las aplicaciones que soporta un

backbone MPLS. Es necesario saber que al momento de tener una aplicación de VPN MPLS, los paquetes atravesando la nube llevan dos etiquetas siendo la primera aquella que lleva la dirección del ruteador de frontera que será su salto siguiente y la segunda aquella que le indica como el dispositivo de frontera alcanzado debe enviar la información al dispositivo ubicado en el lado de un cliente.

Para cuestiones de ingeniería de tráfico se logra entender que un túnel LSP ayuda a evitar la sobrecarga de información en una red, logrando de esa manera que se tengan rutas o caminos específicos para la información de alta importancia y caminos opcionales para las aplicaciones de baja prioridad, dando lugar a que el mismo trato que se le puede dar al tráfico en entornos reales mediante técnicas de calidad de servicio y técnicas de gestión de ancho de banda pueda ser dado también al tráfico de menor volumen en los entornos de simulación logrando de esa manera que la herramienta de emulación sea útil para implementaciones reales habiendo primero analizado los resultados obtenidos.

5 ANÁLISIS DE COSTOS

Un proveedor de servicios que ingresa al mercado queriendo proporcionar servicios de transmisión de datos y competir con los proveedores ya establecidos, debe ingresar con una fuerte inversión, ya que los costos de implementación de una red de transmisión MPLS suponen un gasto enorme según lo demande el equipamiento utilizado, el mismo que debe brindar la seguridad necesaria en el transporte de la información.

Para tener una visión de los equipos y el número de tarjetas que se puede necesitar en la implementación de una red con un numero pequeño de nodos y con capacidades de expansión para futuras acciones referentes al crecimiento de la red, se puede realizar una comparación técnica y económica mediante la sección siguiente, donde se presenta una tabla con los requerimientos de los nodos que se necesitan para formar un backbone de estructura IP – MPLS.

5.1 Análisis General de Costos

Para el diseño de la sencilla red MPLS presentada (fig. 3-1), se ha escogido equipamiento de plataforma Cisco 7206VXR como dispositivos de enrutamiento para el *Backbone* MPLS y Cisco 3745 para la conexión de los clientes hacia la red.

Asumiendo una última milla (Fibra Óptica) ya instalada, la tabla 5-1 detalla las tarjetas necesarias tanto para nodos del *Backbone* como para los usuarios de la red de transmisión de datos

Cantidad	Capacidad	Descripcion
2	4E1	Puertos para conexión entre LSRs y LERs (PE0 y P0)
2	4E1	Puertos para conexión entre LSRs del núcleo de red (P0 y P1)
2	4E1	Puertos para conexión entre LSRs y LERs (P1 y PE1)
2	2E1	Puertos para conexión entre LSRs del núcleo de red (P0 y P2)
2	2E1	Puertos para conexión entre LSRs (P0 y P3)
2	2E1	Puertos para conexión entre LSRs (P1 y P2)
2	2E1	Puertos para conexión entre LSRs (P1 y P3)
2	1/2 E1	Puertos para conexión entre Matriz y dispositivo de Frontera a la Red MPLS
2	1/2 E1	Puertos para conexión entre Agencia de Quito y Dispositivo de Frontera a la Red MPLS
2	1/2 E1	Puertos para conexión entre Agencia Cuenca y Dispositivo de Frontera a la Red MPLS

Tabla 5-1.- Puertos Necesarios para la conexión entre nodos y de los clientes al dominio MPLS

Costo de Ruteadores para el Backbone MPLS

El marco referencial de los costos de los equipos que se utilizan para implementar una red MPLS con un numero de nodos igual a 6 y un numero de puertos detallados (tabla 5-1); es presentado a continuación, eligiendo como plataforma de implementación a los dispositivos Cisco ya mencionados anteriormente. Es necesario dar a conocer que los costos presentados son para infraestructura y *hardware* que puede ser utilizado.

Ruteadores del Dominio MPLS				
Codigo Equipo	**Descripcion**	**Cantidad**	**P. Unitario**	**Total**
C7206VXR/400/2FE	7206VXR with NPE-400 and I/O Controller with 2 FE/E Ports	6	17.500,00	105.000
PWR-7200	Cisco 7200 AC Power Supply Option	6	Included	Included
PWR-7200/2	Cisco 7200 Redundant AC Power Supply Option (280W)	6	$3.000,00	$18.000,00
CAB-AC	Power Cord,110V	6	Included	Included
S72AESK9-12409T	Cisco 7200 IOS ADVANCED ENTERPRISE SERVICES	6	$6.500,00	$39.000,00
PA-4T+	4 Port Serial Port Adapter, Enhanced	6	$4.500,00	$27.000,00
CAB-V35MT	V.35 Cable, DTE, Male, 10 Feet	19	$100,00	$1.900,00
MEM-I/O-FLD64M	Cisco 7200 I/O PCMCIA Flash Disk Memory, 64MB (default)	6	Included	Included
MEM-NPE-400-256MB	256MB Memory for NPE-400 in 7200 Series	6	Included	Included
CON-SNTP-7206	Cisco 7206 SMARTnet Premium Maintenance	6	$7.278,00	$43.668,00
Total			**$38.878,00**	**$234.568,00**

Tabla 5-2.- Costo de Ruteadores del Backbone MPLS

En la tabla 5-2 de detallan los costos del equipamiento con capacidades MPLS que debe ser ubicado en el *Backbone*. Se da a conocer el costo del equipamiento de la infraestructura de la red *(Hardware)* así como el costo de cada uno de los elementos que contiene el equipo de la plataforma Cisco 7200 así como también el costo de mantenimiento que ofrece el distribuidor del producto.

Costo de Ruteadores para el envío de información de los Clientes

En la tabla 5-3 se detalla el costo de los dispositivos de red que una empresa con 3 sitios debe utilizar para la conexión hacia la red MPLS y para la transmisión de información. El Equipo ofrecido cuenta con un sistema operativo con diversas características y funcionalidades y apto para servicios

empresariales, un módulo de red serial con 4 puertos de los cuales es utilizado 1 puerto quedando los demás para futuras expansiones.

Ruteadores de los clientes				
Codigo Equipo	**Descripción**	**Cantidad**	**P. Unitario**	**Total**
CISCO3845	3845 w/AC PWR,2FE,1SFP,4NME,4HWIC, IP Base, 64F/256D	3	$13.000,00	$39.000,00
S384ESK9-12416	Cisco 3845 ENTERPRISE SERVICES	3	$2.000,00	$6.000,00
NM-4T	4-Port Serial Network Module	3	$3.000,00	$9.000,00
CAB-V35MT	V.35 Cable, DTE, Male, 10 Feet	3	$100,00	$300,00
PWR-3845-AC	Cisco 3845 AC power supply	3	Included	Included
CAB-AC	Power Cord,110V	3	Included	Included
ROUTER-SDM	Device manager for routers	3	Included	Included
MEM3800-256D-INCL	256BM SDRAM default memory for 3800	3	Included	Included
MEM3800-64CF-INCL	64MB Cisco 3800 Compact Flash Memory Default	3	Included	Included
CON-SNTP-3845	SMARTNET 24X7X4 3845 w/AC PWR,2FE,1S (Maintenance)	3	$3.319,00	$9.957,00
Total			$21.419,00	**$ 64.257**

Tabla 5-3.- Costo de Ruteadores de los Clientes

Cabe recalcar que los costos del equipamiento se realizaron en base a la plataforma Cisco 3800 ya que es la que ocupa el lugar de la plataforma 3700, la misma que fue utilizada únicamente por propósitos de simulación dado que, *Dynamips* no soporta aun la plataforma Cisco 3800. El cliente puede optar por utilizar los dispositivos que crea conveniente. Se ha escogido la plataforma de dispositivos Cisco 3745 dado que es una de las plataformas que soporta el simulador como ya se hizo mención y además soporta las funcionalidades en cuanto al manejo de gran cantidad de información que puede procesar un cliente. La tabla 5-3 también indica el costo de mantenimiento de los equipos por parte de un distribuidor de plataformas Cisco, el mismo que se les ofrece a los clientes según lo requieran.

CONCLUSIONES Y RECOMENDACIONES

Conclusiones

- Se presenta una solución a los proveedores de servicios que deseen migrar sus redes actuales a un ambiente seguro mediante la implementación de redes MPLS para entablar enlaces ya sean de voz y datos de aquellos clientes que deseen enviar información a través de estas redes.

- El refuerzo al estudiar la tecnología MPLS se valida con la implementación de aplicaciones que puede soportar un *Backbone* con tecnologías de nueva generación, conjuntamente con el uso de herramientas que permiten ejecutar directamente los sistemas operativos de plataformas de Cisco y además permiten que se simulen entornos como si se estuviera trabajando en un ambiente real.

- Para el diseño de una red MPLS se consideran aspectos como: redundancia de rutas, equipos con sistema operativo adecuado para el soporte de la tecnología y eficientes en lo que corresponde al manejo del gran volumen de información que circula por un núcleo de la red de un proveedor.

- Al implementar los servicios de VPNs en MPLS no es necesario contar con direccionamiento global o público para la formación de la conexión punto a punto ya que este tipo de tecnologías, crea las VPN en base a instancias de enrutamiento y envío (VRF), por lo cual un

cliente puede conservar su esquema de direccionamiento privado sin necesidad de realizar traducción de direcciones privadas a públicas (NAT).

- Los túneles de ingeniería de trafico MPLS son de mucha importancia en lo que respecta a la señalización de caminos específicos por los que la información debe viajar, llegando a su destino por una vía mas rápida o un enlace menos congestionado.

- La calidad de servicio en una red mejora el desempeño de la misma y ayuda a priorizar a las aplicaciones que se ejecutan en los sitios de los clientes y que posteriormente atraviesan una nube MPLS. Gracias a las herramientas de QoS se logra que los usuarios respeten las condiciones de transmisión que se pueden estipular en un contrato, manejando las mismas con técnicas que preserven a la información de las pérdidas, retardos, etc.

- La conexión de una Matriz con sus agencias puede verse como una topología lógica en forma de estrella, lo que hace pensar a los clientes que se cuenta con los enlaces dedicados

- Aplicaciones tales como: servicio de correo electrónico, mensajería instantánea, transferencia de archivos, voz, video, administración remota, etc., deben ser correctamente gestionados por un proveedor de servicios de transporte MPLS para evitar la sobrecarga en el núcleo de red sirviendo además eficientemente a las aplicaciones de misión crítica

Recomendaciones.

- Para comprender profundamente una tecnología de nueva generación, es recomendable conocer y tener en claro las funciones de los elementos que son partícipes y que definen la topología de una red teniendo en cuenta además la ubicación que deben tener los mismos.

- Para un cliente que desee entablar sus redes de voz y datos distantes geográficamente, se recomienda que un proveedor de servicios aplique redes privadas virtuales como solución mas viable tanto técnica como económicamente en la conexión punto a punto de las sucursales logrando de esa manera que el cliente sienta que posee los enlaces físicos.

- Es recomendable que el tráfico enviado por los usuarios sea gestionado en el *Backbone* distribuyendo el mismo en clases de servicio diferenciando las mismas por sus niveles de prioridad logrando que tengan ventajas las aplicaciones de misión crítica.

- Generalmente es recomendable que los proveedores de servicios de redes de telecomunicaciones ofrezcan una red unificada en su *Backbone* mediante el protocolo MPLS simplificando su operación y mantenimiento, además potencializando su inversión en redes heredadas (ATM, Frame - Relay, PPP, etc.) al multiplicar el ancho de banda soportado.

TRABAJO FUTURO

Como trabajo futuro se deja abierto en el estudio de estas nuevas tecnologías, los campos de seguridad en redes MPLS y el uso de servicios diferenciados *(diffserv MPLS)*, así como la ejecución de la tecnología sobre los diferentes medios de transporte en capa de enlace de datos (ATM, Frame Relay, SONET, etc), los mismos que se trabajarán y se mejoraran gracias a la continuidad en los avances y mejoramientos de MPLS, esperando recibir el aporte y las criticas constructivas de los lectores para avanzar en el campo investigativo concerniente a las tecnologías de redes y al uso de herramientas de simulación que involucren el desempeño de un mundo virtual trabajando conjuntamente con un mundo real.

BIBLIOGRAFIA

[1] CANALIS MARIA SOL. "Multiprotocol Label Switching – Una Arquitectura de Backbone para la Internet del Siglo XXI". Año: xxxx. Corrientes – Argentina. http://exa.unne.edu.ar/depar/areas/informatica/SistemasOperativos/libmpls.PDF.

[2] DE GHEIN LUC. "Cisco MPLS Fundamentals". Noviembre 2006. http://www.net130.com/2004/6-24/03834.html.

[3] OLOF PETERSON JOHAN MARTÍN. "MPLS Based Recovery Mechanisms". Mayo 2005. Oslo. http://folk.uio.no/johanmp/MPLS%20Based%20Recovery%20Mechanisms.pdf.

[4] INTERNATIONAL ENGINEERING CONSORTIUM. "Multiprotocol Label Switching – MPLS". http://www.iec.org.

[5] CISCO SYSTEMS. "Cisco IOS Multiprotocol Label Switching Configuration Guide". http://www.cisco.com/application/pdf/en/us/guest/products/ps6350/c2001/ccmigration_09186a0080789b24.pdf

[6] ZAMORA HUGO. "Implementación de Redes MPLS – VPNs Casos de Estudio". Año 2002. México. http://www.cudi.edu.mx/primavera2002/presentaciones/MPLSVPN.pdf

[7] BRITTAIN PAUL, FARREL ADRIAN. "MPLS Virtual Private Networks". Noviembre 2000. http://www.cse.iitb.ac.in/~varsha/allpapers/network-misc/mplsvpns.pdf.

[8] SANCHEZ LÓPEZ SERGIO. “Interconnection of IP / MPLS Networks through ATM and Optical Backbones using PNNI Protocols”. Junio 2003. Cataluña – España. http://www.tdx.cesca.es/TESIS_UPC/AVAILABLE/TDX-0729104-125109//TESI.pdf

[9] BELZARENA PABLO. “Ingeniería de tráfico en Redes MPLS aplicando la Teoría de Grandes Desviaciones”. Año 2003. Montevideo – Uruguay. http://iie.fing.edu.uy/publicaciones/2003/Bel03/Bel03.pdf

[10] ULLOA DE SOUZA ALEJANDRO. “Análisis, Diseño de una Subred de Comunicaciones Metro Ethernet Basado en la Tecnología MPLS Aplicada a Estudio de la Integración de Servicios”. Capítulo 2. – “Metro Ethernet y MPLS”. Escuela Superior Politécnica Nacional. Marzo 2007. Quito – Ecuador.

[11] CISCO SYSTEMS. “Border Gateway Protocol”. http://www.cisco.com/en/US/products/ps6647/products_ios_protocol_option_home.html

[12] CISCO SYSTEMS. “Configuring BGP”. http://www.cisco.com/en/US/products/sw/iosswrel/ps1835/products_configuration_guide_cha6pter09186a00800ca763.html

[13] REQUEST FOR COMMENTS – RFC 2283. “Multiprotocol BGP”. http://www.ietf.org/rfc/rfc2283.txt.

[14] CISCO SYSTEMS. “Cisco IOS IP Configuration Guide – Configuring OSPF”. http://www.cisco.com/en/US/products/sw/iosswrel/ps1835/products_configuration_guide_chapter09186a00800b3f2e.html

[15] CISCO SYSTEMS. “Cisco Networking Academy”. Module 1 – Networking Basics.

[16] CISCO SYSTEMS. “Cisco Networking Academy”. Module 2 – Routing Basics.

[17] CISCO SYSTEMS. “Cisco Networking Academy”. Module 3 – Intermediate Routing and Switching.

[18] CISCO SYSTEMS. “Cisco Networking Academy”. Module 4 – WAN Technologies.

[19] CISCO SYSTEMS. “Cisco 7206 Installation and Configuration Guide”. http://www.cisco.com/application/pdf/en/us/guest/products/ps348/c2001/ccmigration_09186a0080201fb9.pdf

[20] PAGINA WEB DYNAMIPS
http://www.ipflow.utc.fr/index.php/Cisco_7200_Simulator

[21] ODOM WENDELL. “CCNA Intro Exam Certification Guide – Cisco IOS”. Año 2004.
http://www.net130.com/2004/10-29/171713.html

[22] CISCO SYSTEMS. “Configuring MPLS Layer 3 VPNs”. Mayo 2005.
http://www.cisco.com/univercd/cc/td/doc/product/software/ios124/124cg/hmp_c/part20/mpbbk4.pdf

[23] APCAR JEFF, GUICHARD JIM, PEPELNJAK IVAN. “MPLS and VPN Architectures”. Volumen I. Marzo 2001.
http://www.ciscopress.com/bookstore/product.asp?isbn=1587054361.

[24] APCAR JEFF, GUICHARD JIM, PEPELNJAK IVAN. “MPLS and VPN Architectures”. Volumen II. Junio 2003. http://www.ciscopress.com.

[25] CISCO SYSTEMS. “MPLS Traffic Engineering”.
http://www.cisco.com/univercd/cc/td/doc/product/software/ios120/120newft/120t/120t7/te120_7t.pdf.

[26] CISCO SYSTEMS. “MPLS Traffic Engineering and Enhancements”. http://www.cisco.com/univercd/cc/td/doc/product/software/ios121/121newft/121t/121t3/traffeng.pdf

[27] CISCO SYSTEMS. “Implementing and MPLS VPN over TE Tunnels”. http://www.cisco.com/warp/public/105/mplsvpnte.pdf

[28] CISCO SYSTEMS. “Cisco IOS Quality of Service Solutions Configuration Guide”. http://www.cisco.com/application/pdf/en/us/guest/products/ps6350/c2001/ccmigration_09186a0080789b65.pdf

[29] CISCO SYSTEMS. “Classifying Network Traffic Using NBAR”. http://www.cisco.com/univercd/cc/td/doc/product/software/ios124/124tcg/tqos_c/part_05/qstclpkt.pdf

[30] CISCO PRESS. “Introduction to IP Qos”. http://www.net130.com/2004/10-29/173535.html

[31] ODEM WENDELL, CAVANAUGH MICHAEL. “IP Telephony Self – Study, Cisco QoS Exam Certification Guide”. Edición 2. Octubre 2004. http://www.net130.com/ccna.

[32] CISCO PRESS. “Building Cisco Remote Access Network”. http://www.cisco.com/web/learning/le3/current_exams/642-821.html.

GLOSARIO

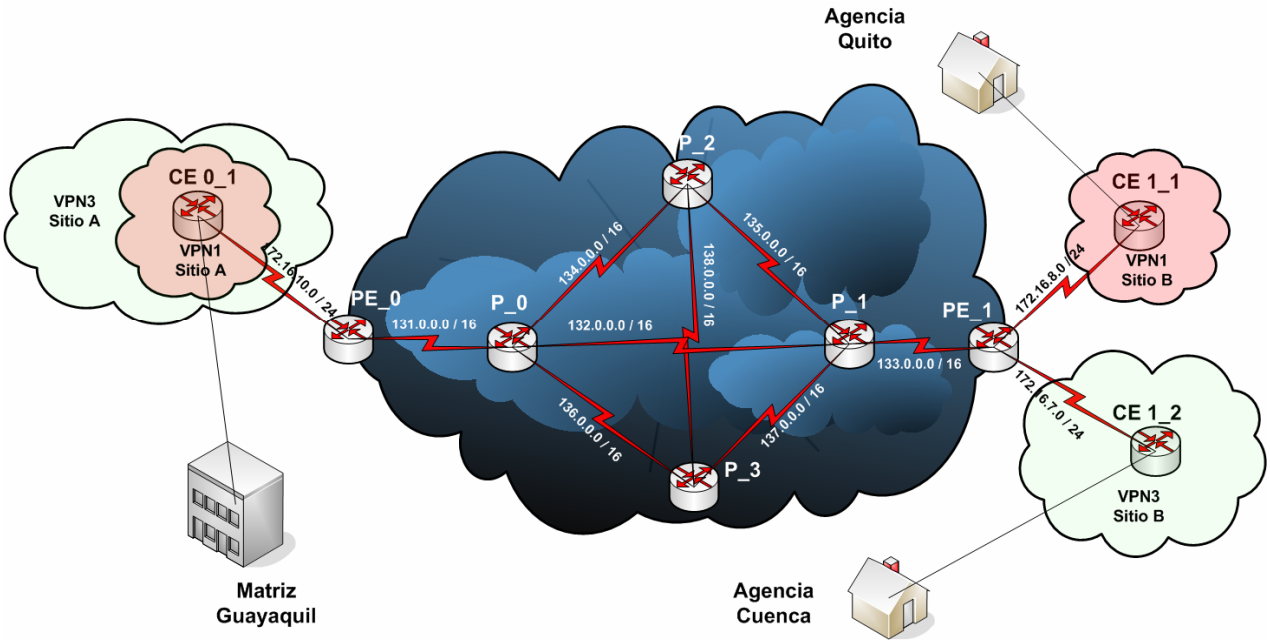
AIM – Advanced Integration Module. Modulo de Integración avanzado.
AS – Autonomous System. Sistema Autónomo.
ATM – Asynchronous Transfer Mode. Modo de Transferencia Asíncrono
BDR – Backup Designed Router. Ruteador designado de respaldo.
BGP – Border Gateway Protocol. Protocolo de puerta de frontera.
CB - Marking – Class Based Marking. Marcado basado en clases
CBWFQ – Class Based Weighted Fair Queueing.
CE – Customer Equipment. Equipamiento en lado del Cliente
CEF – Cisco Express Forwarding. Envío expreso de Cisco.
CIR – Committed Information Rate. Tasa de información Obligatoria
CLI – Command Line Interface. Interface de línea de comandos
CQ – Custom Queuing.
DCBWFQ – Distributed Class Based Weighted Fair Queueing.
DR – Designed Router. Ruteador designado
DWFQ – Distributed Weighted Fair Queueing.
E-BGP – Exterior Border Gateway Protocol. BGP para exteriores.
EIGRP – Enhanced Interior Gateway Routing Protocol. IGRP reforzado
EXP – Campo “Experimental” Usado por MPLS para Calidad Servicio.
FEC – Forwarding Equivalence Class. Clase equivalente de envío.
FIB – Forwarding Information Base. Base de información de envío
FIFO – First Input First Output. Primero en entrar, primero en salir.
FTN – FEC To NHLFE.
FTP – File Transfer Protocol. Protocolo de transferencia de archivos.
HDSM – High Density Service Module. Módulo de servicio de alta densidad
IANA – Internet Assignation Numbers Association.
I-BGP – Interior Border Gateway Protocol. BGP para interiores.
IETF – Internet Engineering Task Force. Fuerza de tareas de ingeniería de Internet
IGRP – Interior Gateway Routing Protocol. Protocolo de enrutamiento de interiores
IHL o HLEN– Internet Header Length. Longitud de la cabecera de capa de red
ILM – Incoming Label Map. Mapa de etiquetas entrantes.
IOS – Internetworking Operative System. Sistema operativo de internetwork
IP – Internet Protocol. Protocolo de Internet.
IPS – Intrusion Prevention System. Sistema de prevención de intrusos
IPX – Internet Protocol Exchange. Intercambio de Protocolo de Internet
IS – IS – Intersystem – Intersystem. Protocolo de enrutamiento inter - sistemas
ISP – Internet Service Provider. Proveedor de servicios de internet.
LAN – Local Area Network. Red de area local.

LDP – Label Distribution Protocol. Protocolo de distribución de etiquetas.
LER – Label Edge Router. Ruteador de frontera de etiquetas
LIB – Label Information Base. Base de informacion de etiquetas.
LLQ – Low Latency Queuing. Encolamiento de baja latencia
LSA – Link State Advertisement. Publicación de estado de enlace.
LSP – Label Switched Path. Ruta conmutada de etiquetas.
LSR – Label Switch Router. Ruteador conmutador de etiquetas
LSU – Link State Updates. Actualizaciones de estado de enlace.
MAC – Media Access Control. Control de acceso al medio
MP-BGP – Multi Protocol Border Gateway Protocol. Extensión Multiprotocolo para BGP.
MPLS – Multiprotocol Label Switching.
MQC – Modular QoS CLI.
NAT – Network Address Translation. Traducción de dirección de red
NBAR – Network Based Application Recognition. Reconocimiento de aplicaciones basadas en la red.
NBMA – Non Broadcast Multi Access. Multi acceso sin *broadcast.*
NHLFE – Next Hop Label Forwarding Entry. Entrada de envoi de etiquetas al siguiente salto.
NLRI – Network Layer Reachable Information. Información alcanzable de capa de red.
NM – Network Module. Modulo de red
NPE – Network Processing Engine. Motor de procesamiento de red.
NSE – Network Service Engine. Motor de servicio de red.
OIR – Online Insertion and Removal. Inserción y remoción en línea.
OSPF – Only Shortest Path First. Protocolo de enrutamiento de únicamente la ruta mas corta primero.
P – Provider. Ruteador del proveedor.
PDU – Protocol Data Unit. Unidad de datos de protocolo.
PE – Provider Edge. Ruteador del frontera al proveedor.
PPP -- Point to Point Protocol. Protocolo de enlace punto a punto
PQ – Priority Queuing. Encolamiento de prioridad.
QoS – Quality of Service. Calidad de servicio.
RD – Route Distinguisher. Distinguidor de ruta.
RFC – Request For Comment. Petición para comentarios.
RIP – Routing Information Protocol. Protocolo de información de enrutamiento.
RSVP – Resource Reservation Protocol. Protocolo de reserve de recursos.
RU – Rack Mountable.
S – “Stack”. Campo de Pila Usado por MPLS.
SNPA – Sub Network Point Added. Punto añadido de sub – red.
TCP/IP – Transport Control Protocol / Internet Protocol. Protocolo de control de transporte / Protocolo de internet.
TLV – Type, Length, Value. Tipo, longitud, valor.

TOS – Type of Service. Tipo de servicio.
TTL – Time To Live. Tiempo de existencia.
UDP -- User Datagram Protocol. Protocolo de datagrama de usuario.
VPN – Virtual Private Network. Red privada virtual.
VRF – VPN Routing and Forwarding Instances. Instancias de enrutamiento y envío VPN
WAN - Wide Area Network. Red de area amplia.
WFQ – Weighted Fair Queuing. Encolamiento de igualdad de pesos.
WIC – WAN Interface Card. Tarjeta de interfaz WAN.
WRED – Weighted Random Early Detection. Detección tempana aleatoria por pesos.

ANEXOS

Configuraciones de cada unos de los dispositivos que representan la red y los servicios que se ofrecen



ANEXO A

Archivo de Configuración que define la topología de la Red MPLS y especifica las conexiones de los Clientes a la Nube

A continuación se presenta el archivo de configuración de los elementos que representan la nube MPLS así como las conexiones desde los clientes hacia la red.

```
##########
#MPLS Lab#
##########

ghostios=true
sparsemem=true

[localhost]

    ############################################################
    #Especificación de los Router que conforman la Red MPLS#
    ############################################################

    [[7200]]
    image = /usr/local/bin/Cisco7200/image7200.bin

    npe = npe-400
    midplane = vxr
    ram = 256

    [[ROUTER PE0]]
    slot1 = PA-4T
    s1/3 = P0 s1/3
    s1/1 = CE0_1 s1/1
    s1/2 = CE0_2 s1/2

    [[ROUTER P0]]
    slot1 = PA-4T
    s1/0 = P2 s1/0
    s1/1 = P3 s1/1
    s1/2 = P1 s1/2

    [[ROUTER P2]]
    slot1 = PA-4T
    s1/1 = P1 s1/1
    s1/2 = P3 s1/2

    [[ROUTER P3]]
    slot1 = PA-4T
    s1/0 = P1 s1/0
```

```
[[ROUTER P1]]
slot1 = PA-4T
s1/3 = PE1 s1/3

[[ROUTER PE1]]
slot1 = PA-4T
s1/1 = CE1_1 s1/1
s1/2 = CE1_2 s1/2


################################################
#Especificación de los Routers de los Clientes#
################################################

[[3745]]
image = /usr/local/bin/Cisco3745/image3745.bin
ram=128

[[ROUTER CE0_1]]
model = 3745
slot1 = NM-4T
f0/0 = NIO_linux_eth:eth0

[[ROUTER CE0_2]]
model = 3745
slot1 = NM-4T
f0/0 = NIO_linux_eth:eth1

[[ROUTER CE1_1]]
model = 3745
slot1 = NM-4T
f0/0 = NIO_linux_eth:eth2

[[ROUTER CE1_2]]
model = 3745
slot1 = NM-4T
f0/0 = NIO_linux_eth:eth3
```

ANEXO B

Configuración de los Dispositivos de la Red MPLS

Anexo B-1.- Provider Edge 0 (PE0)

```
Provider_Edge_0>en
Password:
Provider_Edge_0#show run
Provider_Edge_0#show running-config
Building configuration...

Current configuration : 5998 bytes
!
version 12.4
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Provider_Edge_0
!
boot-start-marker
boot-end-marker
!
enable secret 5 $1$g3A/$J9fPiH9XLFdyDYGSPfRdt/
!
no aaa new-model
!
!
ip cef
!
!
ip vrf vpn1
 rd 100:110
 route-target export 100:1000
 route-target import 100:1000
!
ip vrf vpn2
 rd 100:120
 route-target export 100:2000
 route-target import 100:2000
!
ip vrf vpn3
 rd 100:130
 route-target export 100:3000
 route-target import 100:3000
!
ip vrf vpn4
 rd 100:140
 route-target export 100:4000
 route-target import 100:4000
!
mpls traffic-eng tunnels
!
!
```

```
!
class-map match-all video_in
 match protocol rtp video
class-map match-all voice_in
 match protocol rtp audio
class-map match-all ftp_in
 match protocol ftp
class-map match-all mail_in
 match protocol smtp
class-map match-all telnet_in
 match protocol telnet
!
!
policy-map output_map
 class video_in
  bandwidth percent 20
 class voice_in
  priority percent 25
 class ftp_in
  bandwidth percent 6
 class mail_in
  bandwidth percent 12
 class telnet_in
  bandwidth percent 8
policy-map input_map
 class voice_in
  set ip precedence 5
  police cir 128000
    conform-action transmit
    exceed-action transmit
    violate-action transmit
 class video_in
  set ip precedence 4
  police cir 128000
    conform-action transmit
    exceed-action transmit
    violate-action transmit
 class mail_in
  set ip precedence 3
  police cir 64000
    conform-action transmit
    exceed-action set-prec-transmit 0
    violate-action drop
 class telnet_in
  set ip precedence 2
  police cir 64000
    conform-action transmit
    exceed-action set-prec-transmit 0
    violate-action drop
```

```
 class ftp_in
  set ip precedence 1
  police cir 64000
    conform-action transmit
    exceed-action set-prec-transmit 0
    violate-action drop
 class class-default
  police cir 64000
    conform-action transmit
    exceed-action transmit
    violate-action drop
!
!
!
!
interface Loopback0
 ip address 11.11.11.11 255.255.255.255
 no clns route-cache
!
interface Tunnel1
 ip unnumbered Loopback0
 mpls label protocol ldp
 mpls ip
 no clns route-cache
 tunnel destination 17.17.17.17
 tunnel mode mpls traffic-eng
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 1 1
 tunnel mpls traffic-eng bandwidth  2048
 tunnel mpls traffic-eng path-option 1 dynamic
 no routing dynamic
!
interface Tunnel7
 description "Tunel para la VRF_sitio_A de la VPN4"
 bandwidth 512
 ip vrf forwarding vpn4
 ip address 200.200.202.1 255.255.255.0
 qos pre-classify
 mpls ip
 no clns route-cache
 tunnel source Serial1/2
 tunnel destination 172.16.9.2
 tunnel vrf vpn2
 service-policy input input_map
!
```

```
interface Tunnel8
 description "Tunel para la VRF_sitio_A de la VPN3"
 bandwidth 512
 ip vrf forwarding vpn3
 ip address 200.200.200.1 255.255.255.0
 qos pre-classify
 mpls ip
 no clns route-cache
 tunnel source Serial1/1
 tunnel destination 172.16.10.2
 tunnel vrf vpn1
 service-policy input input_map
!
interface FastEthernet0/0
 bandwidth 8192
 ip address 131.0.0.2 255.255.0.0
 duplex half
 mpls label protocol ldp
 mpls ip
 mpls traffic-eng tunnels
 no clns route-cache
 service-policy output output_map
 ip rsvp bandwidth 8000
!
interface Serial1/0
 no ip address
 shutdown
 serial restart-delay 0
 no clns route-cache
!
interface Serial1/1
 description "Conexion a la VRF_sitio_A de la VPN1"
 bandwidth 1024
 ip vrf forwarding vpn1
 ip address 172.16.10.1 255.255.255.0
 mpls ip
 serial restart-delay 0
 clock rate 2016000
 no clns route-cache
 service-policy input input_map
 service-policy output output_map
!
interface Serial1/2
 description "Conexion a la VRF_sitio_A de la VPN2"
 bandwidth 1024
 ip vrf forwarding vpn2
 ip address 172.16.9.1 255.255.255.0
 mpls ip
 serial restart-delay 0
 clock rate 2016000
 no clns route-cache
 service-policy input input_map
 service-policy output output_map
```

```
!
interface Serial1/3
 description "Conexion al Provider_0"
 bandwidth 8192
 no ip address
 encapsulation ppp
 shutdown
 mpls label protocol ldp
 mpls ip
 mpls traffic-eng tunnels
 serial restart-delay 0
 no clns route-cache
 service-policy output output_map
 ip rsvp bandwidth 4096
!
router ospf 1
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
 log-adjacency-changes
 network 11.11.11.11 0.0.0.0 area 0
 network 131.0.0.0 0.0.255.255 area 0
!
router bgp 100
 no synchronization
 bgp log-neighbor-changes
 neighbor 17.17.17.17 remote-as 100
 neighbor 17.17.17.17 update-source Loopback0
 no auto-summary
 !
 address-family vpnv4
 neighbor 17.17.17.17 activate
 neighbor 17.17.17.17 send-community both
 exit-address-family
 !
 address-family ipv4 vrf vpn4
 redistribute connected
 redistribute static
 no synchronization
 exit-address-family
 !
 address-family ipv4 vrf vpn3
 redistribute connected
 redistribute static
 no synchronization
 exit-address-family
 !
```

```
 !
 address-family ipv4 vrf vpn2
 redistribute connected
 redistribute static
 no synchronization
 exit-address-family
 !
 address-family ipv4 vrf vpn1
 redistribute connected
 redistribute static
 no synchronization
 exit-address-family
!
ip route vrf vpn1 172.16.10.0 255.255.255.0 Serial1/1
ip route vrf vpn1 192.168.10.0 255.255.255.0 172.16.10.2
ip route vrf vpn2 172.16.9.0 255.255.255.0 Serial1/2
ip route vrf vpn2 192.168.9.0 255.255.255.0 172.16.9.2
ip route vrf vpn3 172.16.10.0 255.255.255.0 Tunnel8
ip route vrf vpn3 192.168.10.0 255.255.255.0 200.200.200.2
ip route vrf vpn3 200.200.200.0 255.255.255.0 Tunnel8
ip route vrf vpn4 172.16.9.0 255.255.255.0 Tunnel7
ip route vrf vpn4 192.168.9.0 255.255.255.0 200.200.202.2
ip route vrf vpn4 200.200.202.0 255.255.255.0 Tunnel7
!
no ip http server
!
!
!
!
!
!
control-plane
!
!
!
!
dial-peer cor custom
!
!
!
!
gatekeeper
 shutdown
!
banner motd ^CCRouter de Frontera al Dominio MPLS^C
!
line con 0
 password cisco
 login
 stopbits 1
```

Anexo B-2.- Provider 0 (P0)

```
Provider_0#
Provider_0#show running-config
Building configuration...

Current configuration : 3395 bytes
!
version 12.4
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Provider_0
!
boot-start-marker
boot-end-marker
!
enable secret 5 $1$g3A/$J9fPiH9XLFdyDYGSPfRdt/
!
no aaa new-model
!
!
ip cef
!
!
mpls traffic-eng tunnels
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
```

```
!
!
class-map match-all telnet
 match mpls experimental topmost 2
class-map match-all mail
 match mpls experimental topmost 3
class-map match-all transfer
 match mpls experimental topmost 1
class-map match-all video
 match mpls experimental topmost 4
class-map match-all voice
 match mpls experimental topmost 5
!
!
policy-map output_map
 class voice
  priority percent 25
 class video
  bandwidth percent 20
 class mail
  bandwidth percent 12
  random-detect
 class telnet
  bandwidth percent 8
  random-detect
 class transfer
  bandwidth percent 6
  random-detect
!
!
!
!
interface Loopback0
 ip address 12.12.12.12 255.255.255.255
 no clns route-cache
!
interface Tunnel1
 ip unnumbered Loopback0
 mpls label protocol ldp
 mpls ip
 no clns route-cache
 tunnel destination 17.17.17.17
 tunnel mode mpls traffic-eng
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 1 1
 tunnel mpls traffic-eng bandwidth  2048
 tunnel mpls traffic-eng path-option 1 dynamic
 no routing dynamic
!
```

```
interface Tunnel4
 ip unnumbered Loopback0
 mpls label protocol ldp
 mpls ip
 no clns route-cache
 tunnel destination 11.11.11.11
 tunnel mode mpls traffic-eng
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 1 1
 tunnel mpls traffic-eng bandwidth  2048
 tunnel mpls traffic-eng path-option 4 dynamic
 no routing dynamic
!
interface FastEthernet0/0
 bandwidth 8192
 ip address 131.0.0.1 255.255.0.0
 duplex half
 mpls label protocol ldp
 mpls ip
 mpls traffic-eng tunnels
 no clns route-cache
 service-policy output output_map
 ip rsvp bandwidth 8000
!
interface Serial1/0
 bandwidth 4096
 ip address 134.0.0.1 255.255.0.0
 encapsulation ppp
 mpls label protocol ldp
 mpls ip
 serial restart-delay 0
 clock rate 4032000
 no clns route-cache
 service-policy output output_map
!
interface Serial1/1
 bandwidth 4096
 ip address 136.0.0.1 255.255.0.0
 encapsulation ppp
 mpls label protocol ldp
 mpls ip
 serial restart-delay 0
 clock rate 4032000
 no clns route-cache
 service-policy output output_map
!
```

```
interface Serial1/2
 bandwidth 8192
 ip address 132.0.0.1 255.255.0.0
 encapsulation ppp
 mpls label protocol ldp
 mpls ip
 mpls traffic-eng tunnels
 serial restart-delay 0
 clock rate 8064000
 no clns route-cache
 service-policy output output_map
 ip rsvp bandwidth 8000
!
interface Serial1/3
 bandwidth 8192
 no ip address
 encapsulation ppp
 shutdown
 mpls label protocol ldp
 mpls ip
 mpls traffic-eng tunnels
 serial restart-delay 0
 clock rate 8064000
 no clns route-cache
 service-policy output output_map
 ip rsvp bandwidth 4096
!
router ospf 1
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
 log-adjacency-changes
 network 12.12.12.12 0.0.0.0 area 0
 network 131.0.0.0 0.0.255.255 area 0
 network 132.0.0.0 0.0.255.255 area 0
 network 134.0.0.0 0.0.255.255 area 0
 network 136.0.0.0 0.0.255.255 area 0
!
!
no ip http server
!
!
!
!
!
!
control-plane
!
!
!
!
dial-peer cor custom
!
!
```

Anexo B-3.- Provider 1 (P1)

```
Provider_1#show running-config
Building configuration...

Current configuration : 3261 bytes
!
version 12.4
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Provider_1
!
boot-start-marker
boot-end-marker
!
enable secret 5 $1$j4/M$jbgKHMILTDY.Xvebg4Vd2/
!
no aaa new-model
!
!
ip cef
!
!
mpls traffic-eng tunnels
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
class-map match-all telnet
 match mpls experimental topmost 2
class-map match-all mail
 match mpls experimental topmost 3
class-map match-all transfer
 match mpls experimental topmost 1
class-map match-all video
 match mpls experimental topmost 4
class-map match-all voice
 match mpls experimental topmost 5
```

```
policy-map output_map
 class voice
  priority percent 25
 class video
  bandwidth percent 20
 class mail
  bandwidth percent 12
  random-detect
 class telnet
  bandwidth percent 8
  random-detect
 class transfer
  bandwidth percent 6
  random-detect
!
!
!
!
interface Loopback0
 ip address 13.13.13.13 255.255.255.255
 no clns route-cache
!
interface Tunnel1
 ip unnumbered Loopback0
 mpls label protocol ldp
 mpls ip
 no clns route-cache
 tunnel destination 17.17.17.17
 tunnel mode mpls traffic-eng
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 1 1
 tunnel mpls traffic-eng bandwidth  2048
 tunnel mpls traffic-eng path-option 1 dynamic
 no routing dynamic
!
interface Tunnel4
 ip unnumbered Loopback0
 mpls label protocol ldp
 mpls ip
 no clns route-cache
 tunnel destination 11.11.11.11
 tunnel mode mpls traffic-eng
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 1 1
 tunnel mpls traffic-eng bandwidth  2048
 tunnel mpls traffic-eng path-option 4 dynamic
 no routing dynamic
!
```

```
!
interface FastEthernet0/0
 no ip address
 shutdown
 duplex half
 no clns route-cache
!
interface Serial1/0
 bandwidth 4096
 ip address 137.0.0.2 255.255.0.0
 encapsulation ppp
 mpls label protocol ldp
 mpls ip
 serial restart-delay 0
 clock rate 4032000
 no clns route-cache
 service-policy output output_map
!
interface Serial1/1
 bandwidth 4096
 ip address 135.0.0.2 255.255.0.0
 encapsulation ppp
 mpls label protocol ldp
 mpls ip
 serial restart-delay 0
 clock rate 4032000
 no clns route-cache
 service-policy output output_map
!
interface Serial1/2
 bandwidth 8192
 ip address 132.0.0.2 255.255.0.0
 encapsulation ppp
 mpls label protocol ldp
 mpls ip
 mpls traffic-eng tunnels
 serial restart-delay 0
 clock rate 8064000
 no clns route-cache
 service-policy output output_map
 ip rsvp bandwidth 8000
!
```

```
!
interface Serial1/3
 bandwidth 8192
 ip address 133.0.0.1 255.255.0.0
 encapsulation ppp
 mpls label protocol ldp
 mpls ip
 mpls traffic-eng tunnels
 serial restart-delay 0
 clock rate 8064000
 no clns route-cache
 service-policy output output_map
 ip rsvp bandwidth 8000
!
router ospf 1
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
 log-adjacency-changes
 network 13.13.13.13 0.0.0.0 area 0
 network 132.0.0.0 0.0.255.255 area 0
 network 133.0.0.0 0.0.255.255 area 0
 network 135.0.0.0 0.0.255.255 area 0
 network 137.0.0.0 0.0.255.255 area 0
!
!
no ip http server
!
!
!
!
!
!
control-plane
!
!
!
!
dial-peer cor custom
!
!
!
!
gatekeeper
 shutdown
!
!
line con 0
 password cisco
 login
 stopbits 1
```

Anexo B-4.- Provider 2 (P2)

```
Provider_2#show running-config
Building configuration...

Current configuration : 2333 bytes
!
version 12.4
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Provider_2
!
boot-start-marker
boot-end-marker
!
enable secret 5 $1$g3A/$J9fPiH9XLFdyDYGSPfRdt/
!
no aaa new-model
!
!
ip cef
!
!
mpls traffic-eng tunnels
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
class-map match-all telnet
 match mpls experimental topmost 2
class-map match-all mail
 match mpls experimental topmost 3
class-map match-all transfer
 match mpls experimental topmost 1
class-map match-all video
 match mpls experimental topmost 4
class-map match-all voice
 match mpls experimental topmost 5
!
```

```
!
policy-map output_map
 class voice
  priority percent 25
 class video
  bandwidth percent 20
 class mail
  bandwidth percent 12
  random-detect
 class telnet
  bandwidth percent 8
  random-detect
 class transfer
  bandwidth percent 6
  random-detect
!
!
!
!
interface Loopback0
 ip address 14.14.14.14 255.255.255.255
 no clns route-cache
!
interface FastEthernet0/0
 no ip address
 shutdown
 duplex half
 no clns route-cache
!
interface Serial1/0
 bandwidth 4096
 ip address 134.0.0.2 255.255.0.0
 encapsulation ppp
 mpls label protocol ldp
 mpls ip
 serial restart-delay 0
 clock rate 4032000
 no clns route-cache
 service-policy output output_map
!
interface Serial1/1
 bandwidth 4096
 ip address 135.0.0.1 255.255.0.0
 encapsulation ppp
 mpls label protocol ldp
 mpls ip
 serial restart-delay 0
 clock rate 4032000
 no clns route-cache
 service-policy output output_map
```

```
!
interface Serial1/2
 bandwidth 4096
 ip address 138.0.0.1 255.255.0.0
 encapsulation ppp
 mpls label protocol ldp
 mpls ip
 serial restart-delay 0
 clock rate 4032000
 no clns route-cache
 service-policy output output_map
!
interface Serial1/3
 no ip address
 shutdown
 serial restart-delay 0
 fair-queue 64 256 37
 no clns route-cache
 ip rsvp bandwidth 2000
!
router ospf 1
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
 log-adjacency-changes
 network 14.14.14.14 0.0.0.0 area 0
 network 134.0.0.0 0.0.255.255 area 0
 network 135.0.0.0 0.0.255.255 area 0
 network 138.0.0.0 0.0.255.255 area 0
!
!
no ip http server
!
!
!
!
!
!
control-plane
!
!
!
!
dial-peer cor custom
!
!
!
!
gatekeeper
 shutdown
!
```

Anexo B-5.- Provider 3 (P3)

```
Provider_3#show running-config
Building configuration...

Current configuration : 2287 bytes
!
version 12.4
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Provider_3
!
boot-start-marker
boot-end-marker
!
enable secret 5 $1$A/ZC$3rUcoyPfVFKD9.vUDsEBI.
!
no aaa new-model
!
!
ip cef
!
!
mpls traffic-eng tunnels
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
class-map match-all telnet
 match mpls experimental topmost 2
class-map match-all mail
 match mpls experimental topmost 3
class-map match-all transfer
 match mpls experimental topmost 1
class-map match-all video
 match mpls experimental topmost 4
class-map match-all voice
 match mpls experimental topmost 5
!
!
```

```
!
policy-map output_map
 class voice
  priority percent 25
 class video
  bandwidth percent 20
 class mail
  bandwidth percent 12
  random-detect
 class telnet
  bandwidth percent 8
  random-detect
 class transfer
  bandwidth percent 6
  random-detect
!
!
!
!
interface Loopback0
 ip address 15.15.15.15 255.255.255.255
 no clns route-cache
!
interface FastEthernet0/0
 no ip address
 shutdown
 duplex half
 no clns route-cache
!
interface Serial1/0
 bandwidth 4096
 ip address 137.0.0.1 255.255.0.0
 encapsulation ppp
 mpls label protocol ldp
 mpls ip
 serial restart-delay 0
 clock rate 4032000
 no clns route-cache
 service-policy output output_map
!
interface Serial1/1
 bandwidth 4096
 ip address 136.0.0.2 255.255.0.0
 encapsulation ppp
 mpls label protocol ldp
 mpls ip
 serial restart-delay 0
 clock rate 4032000
 no clns route-cache
 service-policy output output_map
!
```

```
!
interface Serial1/2
 bandwidth 4096
 ip address 138.0.0.2 255.255.0.0
 encapsulation ppp
 mpls label protocol ldp
 mpls ip
 serial restart-delay 0
 clock rate 4032000
 no clns route-cache
 service-policy output output_map
!
interface Serial1/3
 no ip address
 shutdown
 serial restart-delay 0
 no clns route-cache
!
router ospf 1
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
 log-adjacency-changes
 network 15.15.15.15 0.0.0.0 area 0
 network 136.0.0.0 0.0.255.255 area 0
 network 137.0.0.0 0.0.255.255 area 0
 network 138.0.0.0 0.0.255.255 area 0
!
!
no ip http server
!
!
!
!
!
!
control-plane
!
!
!
!
dial-peer cor custom
!
!
!
!
gatekeeper
 shutdown
!
!
line con 0
 password cisco
 login
 stopbits 1
```

Anexo B-6.- Provider Edge 1 (PE1)

```
Provider_Edge_1#show running-config
Building configuration...

Current configuration : 5877 bytes
!
version 12.4
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Provider_Edge_1
!
boot-start-marker
boot-end-marker
!
enable secret 5 $1$DjYp$cGS4eoIowyKENjGGjakQS.
!
no aaa new-model
!
!
ip cef
!
!
ip vrf vpn1
 rd 100:110
 route-target export 100:1000
 route-target import 100:1000
!
ip vrf vpn2
 rd 100:120
 route-target export 100:2000
 route-target import 100:2000
!
ip vrf vpn3
 rd 100:130
 route-target export 100:3000
 route-target import 100:3000
!
ip vrf vpn4
 rd 100:140
 route-target export 100:4000
 route-target import 100:4000
!
mpls traffic-eng tunnels
!
!
!
!
!
!
```

```
!
!
class-map match-all video_in
 match protocol rtp video
class-map match-all voice_in
 match protocol rtp audio
class-map match-all ftp_in
 match protocol ftp
class-map match-all mail_in
 match protocol smtp
class-map match-all telnet_in
 match protocol telnet
!
!
policy-map output_map
 class video_in
  bandwidth percent 20
 class voice_in
  priority percent 25
 class ftp_in
  bandwidth percent 6
 class mail_in
  bandwidth percent 12
 class telnet_in
  bandwidth percent 8
policy-map input_map
 class voice_in
  set ip precedence 5
  police cir 128000
    conform-action transmit
    exceed-action transmit
    violate-action transmit
 class video_in
  set ip precedence 4
  police cir 128000
    conform-action transmit
    exceed-action transmit
    violate-action transmit
 class mail_in
  set ip precedence 3
  police cir 64000
    conform-action transmit
    exceed-action set-prec-transmit 0
    violate-action drop
 class telnet_in
  set ip precedence 2
  police cir 64000
    conform-action transmit
    exceed-action set-prec-transmit 0
    violate-action drop
```

```
 class ftp_in
  set ip precedence 1
  police cir 64000
    conform-action transmit
    exceed-action set-prec-transmit 0
    violate-action drop
 class class-default
  police cir 64000
    conform-action transmit
    exceed-action transmit
    violate-action drop
!
!
!
!
interface Loopback0
 ip address 17.17.17.17 255.255.255.255
 no clns route-cache
!
interface Tunnel4
 ip unnumbered Loopback0
 mpls label protocol ldp
 mpls ip
 no clns route-cache
 tunnel destination 11.11.11.11
 tunnel mode mpls traffic-eng
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 1 1
 tunnel mpls traffic-eng bandwidth  2048
 tunnel mpls traffic-eng path-option 1 dynamic
 no routing dynamic
!
interface Tunnel7
 description "Tunel para la VRF_sitio_B de la VPN4"
 bandwidth 512
 ip vrf forwarding vpn4
 ip address 200.200.203.1 255.255.255.0
 qos pre-classify
 mpls ip
 no clns route-cache
 tunnel source Serial1/1
 tunnel destination 172.16.8.2
 tunnel vrf vpn1
 service-policy input input_map
```

```
interface Tunnel8
 description "Tunel para la VRF_sitio_B de la VPN3"
 bandwidth 512
 ip vrf forwarding vpn3
 ip address 200.200.201.1 255.255.255.0
 qos pre-classify
 mpls ip
 no clns route-cache
 tunnel source Serial1/2
 tunnel destination 172.16.7.2
 tunnel vrf vpn2
 service-policy input input_map
!
interface FastEthernet0/0
 no ip address
 shutdown
 duplex half
 no clns route-cache
!
interface Serial1/0
 no ip address
 shutdown
 serial restart-delay 0
 no clns route-cache
!
interface Serial1/1
 description "Conexion a la VRF_sitio_B de la VPN1"
 bandwidth 1024
 ip vrf forwarding vpn1
 ip address 172.16.8.1 255.255.255.0
 mpls ip
 serial restart-delay 0
 clock rate 2016000
 no clns route-cache
 service-policy input input_map
 service-policy output output_map
!
interface Serial1/2
 description "Conexion a la VRF_sitio_B de la VPN2"
 bandwidth 1024
 ip vrf forwarding vpn2
 ip address 172.16.7.1 255.255.255.0
 mpls ip
 serial restart-delay 0
 clock rate 2016000
 no clns route-cache
 service-policy input input_map
 service-policy output output_map
!
```

```
!
interface Serial1/3
 description "Conexion al Provider_1"
 bandwidth 8192
 ip address 133.0.0.2 255.255.0.0
 encapsulation ppp
 mpls label protocol ldp
 mpls ip
 mpls traffic-eng tunnels
 serial restart-delay 0
 clock rate 8064000
 no clns route-cache
 service-policy output output_map
 ip rsvp bandwidth 8000
!
router ospf 1
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
 log-adjacency-changes
 network 17.17.17.17 0.0.0.0 area 0
 network 133.0.0.0 0.0.255.255 area 0
!
router bgp 100
 no synchronization
 bgp log-neighbor-changes
 neighbor 11.11.11.11 remote-as 100
 neighbor 11.11.11.11 update-source Loopback0
 no auto-summary
 !
 address-family vpnv4
 neighbor 11.11.11.11 activate
 neighbor 11.11.11.11 send-community both
 exit-address-family
 !
 address-family ipv4 vrf vpn4
 redistribute connected
 redistribute static
 no synchronization
 exit-address-family
 !
 address-family ipv4 vrf vpn3
 redistribute connected
 redistribute static
 no synchronization
 exit-address-family
 !
```

```
 !
 address-family ipv4 vrf vpn2
 redistribute connected
 redistribute static
 no synchronization
 exit-address-family
 !
 address-family ipv4 vrf vpn1
 redistribute connected
 redistribute static
 no synchronization
 exit-address-family
!
ip route vrf vpn1 172.16.8.0 255.255.255.0 Serial1/1
ip route vrf vpn1 192.168.8.0 255.255.255.0 172.16.8.2
ip route vrf vpn2 172.16.7.0 255.255.255.0 Serial1/2
ip route vrf vpn2 192.168.7.0 255.255.255.0 172.16.7.2
ip route vrf vpn3 172.16.7.0 255.255.255.0 Tunnel8
ip route vrf vpn3 192.168.7.0 255.255.255.0 200.200.201.2
ip route vrf vpn3 200.200.201.0 255.255.255.0 Tunnel8
ip route vrf vpn4 172.16.8.0 255.255.255.0 Tunnel7
ip route vrf vpn4 192.168.8.0 255.255.255.0 200.200.203.2
ip route vrf vpn4 200.200.203.0 255.255.255.0 Tunnel7
!
no ip http server
!
!
!
!
!
!
control-plane
!
!
!
!
dial-peer cor custom
!
!
!
!
gatekeeper
 shutdown
!
banner motd ^CCRouter de Frontera al Dominio MPLS^C
!
line con 0
 password cisco
 login
 stopbits 1
```

ANEXO C

Configuración de los dispositivos ubicados en los sitios del cliente

Anexo C-1.- Customer Equipment 0_1 (CE0_1)

```
Customer_0_1#show run
Customer_0_1#show running-config
Building configuration...

Current configuration : 1588 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname Customer_0_1
!
boot-start-marker
boot-end-marker
!
enable secret 5 $1$Tsy3$Nt/qwqRtSDMNW63RizauG1
!
no aaa new-model
memory-size iomem 5
ip cef
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
interface Tunnel8
 bandwidth 512
 ip address 200.200.200.2 255.255.255.0
 tunnel source Serial1/1
 tunnel destination 172.16.10.1
!
```

```
!
interface FastEthernet0/0
 ip address 192.168.10.1 255.255.255.0
 duplex auto
 speed auto
!
interface FastEthernet0/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface Serial1/0
 no ip address
 shutdown
 serial restart-delay 0
!
interface Serial1/1
 bandwidth 1024
 ip address 172.16.10.2 255.255.255.0
 serial restart-delay 0
!
interface Serial1/2
 no ip address
 shutdown
 serial restart-delay 0
!
interface Serial1/3
 no ip address
 shutdown
 serial restart-delay 0
!
ip route 172.16.7.0 255.255.255.0 200.200.200.1
ip route 172.16.8.0 255.255.255.0 172.16.10.1
ip route 172.16.9.0 255.255.255.0 172.16.10.1
ip route 172.16.10.0 255.255.255.0 Serial1/1
ip route 172.16.10.0 255.255.255.0 Tunnel8
ip route 192.168.7.0 255.255.255.0 200.200.200.1
ip route 192.168.8.0 255.255.255.0 172.16.10.1
ip route 192.168.10.0 255.255.255.0 FastEthernet0/0
ip route 200.200.200.0 255.255.255.0 Tunnel8
ip route 200.200.201.0 255.255.255.0 200.200.200.1
!
!
ip http server
no ip http secure-server
!
!
!
!
control-plane
!
```

Anexo C-2.- Customer Equipment 0_2 (CE0_2)

```
Customer_0_2#show running-config
Building configuration...

Current configuration : 1595 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname Customer_0_2
!
boot-start-marker
boot-end-marker
!
enable secret 5 $1$kHYf$1tE5q00Ar.hCFpJZkgSV5.
!
no aaa new-model
memory-size iomem 5
ip cef
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
interface Tunnel7
 bandwidth 512
 ip address 200.200.202.2 255.255.255.0
 tunnel source Serial1/2
 tunnel destination 172.16.9.1
!
```

```
!
interface FastEthernet0/0
 ip address 192.168.9.1 255.255.255.0
 duplex auto
 speed auto
!
interface FastEthernet0/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface Serial1/0
 no ip address
 shutdown
 no fair-queue
 serial restart-delay 0
!
interface Serial1/1
 no ip address
 shutdown
 serial restart-delay 0
!
interface Serial1/2
 bandwidth 1024
 ip address 172.16.9.2 255.255.255.0
 serial restart-delay 0
!
interface Serial1/3
 no ip address
 shutdown
 serial restart-delay 0
!
ip route 172.16.7.0 255.255.255.0 172.16.9.1
ip route 172.16.8.0 255.255.255.0 200.200.202.1
ip route 172.16.9.0 255.255.255.0 Serial1/2
ip route 172.16.9.0 255.255.255.0 Tunnel7
ip route 172.16.10.0 255.255.255.0 172.16.9.1
ip route 192.168.7.0 255.255.255.0 172.16.9.1
ip route 192.168.8.0 255.255.255.0 200.200.202.1
ip route 192.168.9.0 255.255.255.0 FastEthernet0/0
ip route 200.200.202.0 255.255.255.0 Tunnel7
ip route 200.200.203.0 255.255.255.0 200.200.202.1
!
!
ip http server
no ip http secure-server
!
!
!
!
control-plane
!
```

Anexo C-3.- Customer Equipment 1_1 (CE1_1)

```
Customer_1_1#show run
Customer_1_1#show running-config
Building configuration...

Current configuration : 1551 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname Customer_1_1
!
boot-start-marker
boot-end-marker
!
enable secret 5 $1$oIr/$iHwJFvnlwJ5giUG/sj6V/0
!
no aaa new-model
memory-size iomem 5
ip cef
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
interface Tunnel7
 bandwidth 512
 ip address 200.200.203.2 255.255.255.0
 tunnel source Serial1/1
 tunnel destination 172.16.8.1
!
```

```
!
interface FastEthernet0/0
 ip address 192.168.8.1 255.255.255.0
 duplex auto
 speed auto
!
interface FastEthernet0/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface Serial1/0
 no ip address
 shutdown
 no fair-queue
 serial restart-delay 0
!
interface Serial1/1
 bandwidth 1024
 ip address 172.16.8.2 255.255.255.0
 serial restart-delay 0
!
interface Serial1/2
 no ip address
 shutdown
 serial restart-delay 0
!
interface Serial1/3
 no ip address
 shutdown
 serial restart-delay 0
!
ip route 172.16.8.0 255.255.255.0 Serial1/1
ip route 172.16.8.0 255.255.255.0 Tunnel7
ip route 172.16.9.0 255.255.255.0 200.200.203.1
ip route 172.16.10.0 255.255.255.0 172.16.8.1
ip route 192.168.8.0 255.255.255.0 FastEthernet0/0
ip route 192.168.9.0 255.255.255.0 200.200.203.1
ip route 192.168.10.0 255.255.255.0 172.16.8.1
ip route 200.200.202.0 255.255.255.0 200.200.203.1
ip route 200.200.203.0 255.255.255.0 Tunnel7
!
!
ip http server
no ip http secure-server
!
!
!
!
control-plane
!
```

Anexo C-4.- Customer Equipment 1_2 (CE1_2)

```
Customer_1_2#show run
Customer_1_2#show running-config
Building configuration...

Current configuration : 1551 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname Customer_1_2
!
boot-start-marker
boot-end-marker
!
enable secret 5 $1$7Xi7$rw6dqCqK/zadYGrBEaCjL1
!
no aaa new-model
memory-size iomem 5
ip cef
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
interface Tunnel8
 bandwidth 512
 ip address 200.200.201.2 255.255.255.0
 tunnel source Serial1/2
 tunnel destination 172.16.7.1
!
```

```
!
interface FastEthernet0/0
 ip address 192.168.7.1 255.255.255.0
 duplex auto
 speed auto
!
interface FastEthernet0/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface Serial1/0
 no ip address
 shutdown
 no fair-queue
 serial restart-delay 0
!
interface Serial1/1
 no ip address
 shutdown
 serial restart-delay 0
!
interface Serial1/2
 bandwidth 1024
 ip address 172.16.7.2 255.255.255.0
 serial restart-delay 0
!
interface Serial1/3
 no ip address
 shutdown
 serial restart-delay 0
!
ip route 172.16.7.0 255.255.255.0 Serial1/2
ip route 172.16.7.0 255.255.255.0 Tunnel8
ip route 172.16.9.0 255.255.255.0 172.16.7.1
ip route 172.16.10.0 255.255.255.0 200.200.201.1
ip route 192.168.7.0 255.255.255.0 FastEthernet0/0
ip route 192.168.9.0 255.255.255.0 172.16.7.1
ip route 192.168.10.0 255.255.255.0 200.200.201.1
ip route 200.200.200.0 255.255.255.0 200.200.201.1
ip route 200.200.201.0 255.255.255.0 Tunnel8
!
!
ip http server
no ip http secure-server
!
!
!
!
control-plane
!
```

ANEXO D

Mapa de Información de Etiquetas y Envío MPLS en los Dispositivos de Frontera

Anexo D-1.- Provider Edge 0 (PE0)

```
Provider_Edge_0#show mpls for
Provider_Edge_0#show mpls forwarding-table
Local  Outgoing    Prefix           Bytes tag  Outgoing   Next Hop
tag    tag or VC   or Tunnel Id     switched   interface
16     Pop tag     12.12.12.12/32   0          Fa0/0      131.0.0.1
17     23          13.13.13.13/32   0          Fa0/0      131.0.0.1
18     24          14.14.14.14/32   0          Fa0/0      131.0.0.1
19     25          15.15.15.15/32   0          Fa0/0      131.0.0.1
20     Pop tag [T] 17.17.17.17/32   0          Tu1        point2point
21     Pop tag     132.0.0.0/16     0          Fa0/0      131.0.0.1
22     27          133.0.0.0/16     0          Fa0/0      131.0.0.1
23     Pop tag     134.0.0.0/16     0          Fa0/0      131.0.0.1
24     28          135.0.0.0/16     0          Fa0/0      131.0.0.1
25     Pop tag     136.0.0.0/16     0          Fa0/0      131.0.0.1
26     29          137.0.0.0/16     0          Fa0/0      131.0.0.1
27     30          138.0.0.0/16     0          Fa0/0      131.0.0.1
28     Aggregate   172.16.10.0/24[V] 43812
29     Untagged    192.168.10.0/24[V]   \
                                    53672      Se1/1      point2point
30     Aggregate   172.16.9.0/24[V]  43292
31     Untagged    192.168.9.0/24[V] 42752      Se1/2      point2point
32     Untagged    172.16.10.0/24[V] 0          Tu8        point2point
33     Untagged    192.168.10.0/24[V]   \
                                    26132      Tu8        point2point
34     Aggregate   200.200.200.0/24[V]   \
Local  Outgoing    Prefix           Bytes tag  Outgoing   Next Hop
tag    tag or VC   or Tunnel Id     switched   interface
                                    64092
35     Untagged    172.16.9.0/24[V]  0          Tu7        point2point
36     Untagged    192.168.9.0/24[V] 26132      Tu7        point2point
37     Aggregate   200.200.202.0/24[V]   \
                                    64092
38     29      [T] 172.16.8.0/24[V]  0          Tu1        point2point
39     30      [T] 192.168.8.0/24[V] 0          Tu1        point2point
40     31      [T] 172.16.7.0/24[V]  0          Tu1        point2point
41     32      [T] 192.168.7.0/24[V] 0          Tu1        point2point
42     33      [T] 172.16.7.0/24[V]  0          Tu1        point2point
43     34      [T] 192.168.7.0/24[V] 0          Tu1        point2point
44     35      [T] 200.200.201.0/24[V]   \
                                    0          Tu1        point2point
45     36      [T] 172.16.8.0/24[V]  0          Tu1        point2point
46     37      [T] 192.168.8.0/24[V] 0          Tu1        point2point
47     38      [T] 200.200.203.0/24[V]   \
                                    0          Tu1        point2point

[T]     Forwarding through a TSP tunnel.
        View additional tagging info with the 'detail' option
Provider_Edge_0#
```

Anexo D-2.- Provider Edge 1 (PE1)

```
Provider_Edge_1#show mpls forwa
Provider_Edge_1#show mpls forwarding-table
Local  Outgoing    Prefix            Bytes tag  Outgoing   Next Hop
tag    tag or VC   or Tunnel Id      switched   interface
16     Untagged    133.0.0.1/32      0          Se1/3      point2point
17     23          131.0.0.0/16      0          Se1/3      point2point
18     Pop tag [T] 11.11.11.11/32    0          Tu4        point2point
19     24          12.12.12.12/32    0          Se1/3      point2point
20     Pop tag     13.13.13.13/32    0          Se1/3      point2point
21     25          14.14.14.14/32    0          Se1/3      point2point
22     26          15.15.15.15/32    0          Se1/3      point2point
23     Pop tag     132.0.0.0/16      0          Se1/3      point2point
24     28          134.0.0.0/16      0          Se1/3      point2point
25     Pop tag     135.0.0.0/16      0          Se1/3      point2point
26     29          136.0.0.0/16      0          Se1/3      point2point
27     Pop tag     137.0.0.0/16      0          Se1/3      point2point
28     30          138.0.0.0/16      0          Se1/3      point2point
29     Aggregate   172.16.8.0/24[V]  54212
30     Untagged    192.168.8.0/24[V] 43272      Se1/1      point2point
31     Aggregate   172.16.7.0/24[V]  43292
32     Untagged    192.168.7.0/24[V] 42752      Se1/2      po nt2point
33     Untagged    172.16.7.0/24[V]  0          Tu8        point2point
34     Untagged    192.168.7.0/24[V] 26132      Tu8        point2point
35     Aggregate   200.200.201.0/24[V]   \
                                     64092
Local  Outgoing    Prefix            Bytes tag  Outgoing   Next Hop
tag    tag or VC   or Tunnel Id      switched   interface
36     Untagged    172.16.8.0/24[V]  0          Tu7        point2point
37     Untagged    192.168.8.0/24[V] 26132      Tu7        point2point
38     Aggregate   200.200.203.0/24[V]   \
                                     64092
39     28      [T] 172.16.10.0/24[V] 0          Tu4        point2point
40     29      [T] 192.168.10.0/24[V]   \
                                     0          Tu4        point2point
41     30      [T] 172.16.9.0/24[V]  0          Tu4        point2point
42     31      [T] 192.168.9.0/24[V] 0          Tu4        point2point
43     32      [T] 172.16.10.0/24[V] 0          Tu4        point2point
44     33      [T] 192.168.10.0/24[V]   \
                                     0          Tu4        point2point
45     34      [T] 200.200.200.0/24[V]   \
                                     0          Tu4        point2point
46     35      [T] 172.16.9.0/24[V]  0          Tu4        point2point
47     36      [T] 192.168.9.0/24[V] 0          Tu4        point2point
48     37      [T] 200.200.202.0/24[V]   \
                                     0          Tu4        point2point

[T]     Forwarding through a TSP tunnel.
        View additional tagging info with the 'detail' option
Provider_Edge_1#
```

ANEXO E

Tablas de Enrutamiento en los Dispositivos de Frontera

Anexo E-1.- Provider Edge 0 (PE0)

Tabla de Enrutamiento hacia el Backbone

```
Provider_Edge_0#show 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,  2 - 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

     17.0.0.0/32 is subnetted, 1 subnets
O       17.17.17.17 [110/37] via 0.0.0.0, 00:47:01, Tunnel1
O    136.0.0.0/16 [110/36] via 131.0.0.1, 00:47:01, FastEthernet0/0
O    137.0.0.0/16 [110/48] via 131.0.0.1, 00:47:01, FastEthernet0/0
O    138.0.0.0/16 [110/60] via 131.0.0.1, 00:47:01, FastEthernet0/0
C    131.0.0.0/16 is directly connected, FastEthernet0/0
     11.0.0.0/32 is subnetted, 1 subnets
C       11.11.11.11 is directly connected, Loopback0
O    132.0.0.0/16 [110/24] via 131.0.0.1, 00:47:01, FastEthernet0/0
     12.0.0.0/32 is subnetted, 1 subnets
O       12.12.12.12 [110/13] via 131.0.0.1, 00:47:01, FastEthernet0/0
O    133.0.0.0/16 [110/36] via 131.0.0.1, 00:47:01, FastEthernet0/0
     13.0.0.0/32 is subnetted, 1 subnets
O       13.13.13.13 [110/25] via 131.0.0.1, 00:47:02, FastEthernet0/0
O    134.0.0.0/16 [110/36] via 131.0.0.1, 00:47:02, FastEthernet0/0
     14.0.0.0/32 is subnetted, 1 subnets
O       14.14.14.14 [110/37] via 131.0.0.1, 00:47:02, FastEthernet0/0
O    135.0.0.0/16 [110/48] via 131.0.0.1, 00:47:02, FastEthernet0/0
     15.0.0.0/32 is subnetted, 1 subnets
O       15.15.15.15 [110/37] via 131.0.0.1, 00:47:02, FastEthernet0/0
Provider_Edge_0#
```

Tabla de Enrutamiento VPN1 (VRF sitio A)

```
Provider_Edge_0#show ip route vrf vpn1

Routing Table: vpn1
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

B    192.168.8.0/24 [200/0] via 17.17.17.17, 00:46:52
S    192.168.10.0/24 [1/0] via 172.16.10.2
     172.16.0.0/24 is subnetted, 2 subnets
B       172.16.8.0 [200/0] via 17.17.17.17, 00:46:52
C       172.16.10.0 is directly connected, Serial1/1
Provider_Edge_0#
```

Tabla de Enrutamiento VPN2 (VRF sitio A)

```
Provider_Edge_0#show ip route vrf vpn2

Routing Table: vpn2
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

S    192.168.9.0/24 [1/0] via 172.16.9.2
     172.16.0.0/24 is subnetted, 2 subnets
C       172.16.9.0 is directly connected, Serial1/2
B       172.16.7.0 [200/0] via 17.17.17.17, 00:47:09
B    192.168.7.0/24 [200/0] via 17.17.17.17, 00:47:09
Provider_Edge_0#
```

Tabla de Enrutamiento VPN3 (VRF sitio A)

```
Provider_Edge_0#show ip route vrf vpn3

Routing Table: vpn3
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

C    200.200.200.0/24 is directly connected, Tunnel8
B    200.200.201.0/24 [200/0] via 17.17.17.17, 00:47:29
S    192.168.10.0/24 [1/0] via 200.200.200.2
     172.16.0.0/24 is subnetted, 2 subnets
S       172.16.10.0 is directly connected, Tunnel8
B       172.16.7.0 [200/0] via 17.17.17.17, 00:47:29
B    192.168.7.0/24 [200/0] via 17.17.17.17, 00:47:29
Provider_Edge_0#
```

Tabla de Enrutamiento VPN4 (VRF sitio A)

```
Provider_Edge_0#show ip route vrf vpn4

Routing Table: vpn4
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

C    200.200.202.0/24 is directly connected, Tunnel7
B    200.200.203.0/24 [200/0] via 17.17.17.17, 00:47:53
B    192.168.8.0/24 [200/0] via 17.17.17.17, 00:47:53
S    192.168.9.0/24 [1/0] via 200.200.202.2
     172.16.0.0/24 is subnetted, 2 subnets
B       172.16.8.0 [200/0] via 17.17.17.17, 00:47:53
S       172.16.9.0 is directly connected, Tunnel7
Provider_Edge_0#
```

Anexo E-2.- Provider Edge 1 (PE1)

Tabla de Enrutamiento hacia el Backbone

```
Provider_Edge_1#show 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

     17.0.0.0/32 is subnetted, 1 subnets
C       17.17.17.17 is directly connected, Loopback0
O    136.0.0.0/16 [110/48] via 133.0.0.1, 00:41:08, Serial1/3
O    137.0.0.0/16 [110/36] via 133.0.0.1, 00:41:08, Serial1/3
O    138.0.0.0/16 [110/60] via 133.0.0.1, 00:41:08, Serial1/3
O    131.0.0.0/16 [110/36] via 133.0.0.1, 00:41:08, Serial1/3
     11.0.0.0/32 is subnetted, 1 subnets
O       11.11.11.11 [110/37] via 0.0.0.0, 00:41:08, Tunnel4
O    132.0.0.0/16 [110/24] via 133.0.0.1, 00:41:08, Serial1/3
     12.0.0.0/32 is subnetted, 1 subnets
O       12.12.12.12 [110/25] via 133.0.0.1, 00:41:08, Serial1/3
     133.0.0.0/16 is variably subnetted, 2 subnets, 2 masks
C       133.0.0.1/32 is directly connected, Serial1/3
C       133.0.0.0/16 is directly connected, Serial1/3
     13.0.0.0/32 is subnetted, 1 subnets
O       13.13.13.13 [110/13] via 133.0.0.1, 00:41:10, Serial1/3
O    134.0.0.0/16 [110/48] via 133.0.0.1, 00:41:10, Serial1/3
     14.0.0.0/32 is subnetted, 1 subnets
O       14.14.14.14 [110/37] via 133.0.0.1, 00:41:10, Serial1/3
O    135.0.0.0/16 [110/36] via 133.0.0.1, 00:41:10, Serial1/3
     15.0.0.0/32 is subnetted, 1 subnets
O       15.15.15.15 [110/37] via 133.0.0.1, 00:41:10, Serial1/3
Provider_Edge_1#
Provider_Edge_1#
```

Tabla de Enrutamiento VPN1 (VRF sitio B)

```
Provider_Edge_1#show ip route vrf vpn1

Routing Table: vpn1
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

S    192.168.8.0/24 [1/0] via 172.16.8.2
B    192.168.10.0/24 [200/0] via 11.11.11.11, 00:37:40
     172.16.0.0/24 is subnetted, 2 subnets
C       172.16.8.0 is directly connected, Serial1/1
B       172.16.10.0 [200/0] via 11.11.11.11, 00:37:40
Provider_Edge_1#
```

Tabla de Enrutamiento VPN2 (VRF sitio B)

```
Provider_Edge_1#show ip route vrf vpn2

Routing Table: vpn2
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

B    192.168.9.0/24 [200/0] via 11.11.11.11, 00:39:01
     172.16.0.0/24 is subnetted, 2 subnets
B       172.16.9.0 [200/0] via 11.11.11.11, 00:39:01
C       172.16.7.0 is directly connected, Serial1/2
S    192.168.7.0/24 [1/0] via 172.16.7.2
Provider_Edge_1#
```

Tabla de Enrutamiento VPN3 (VRF sitio B)

```
Provider_Edge_1#show ip route vrf vpn3

Routing Table: vpn3
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

B    200.200.200.0/24 [200/0] via 11.11.11.11, 00:39:30
C    200.200.201.0/24 is directly connected, Tunnel8
B    192.168.10.0/24 [200/0] via 11.11.11.11, 00:39:30
     172.16.0.0/24 is subnetted, 2 subnets
B       172.16.10.0 [200/0] via 11.11.11.11, 00:39:30
S       172.16.7.0 is directly connected, Tunnel8
S    192.168.7.0/24 [1/0] via 200.200.201.2
Provider_Edge_1#
```

Tabla de Enrutamiento VPN4 (VRF sitio B)

```
Provider_Edge_1#show ip route vrf vpn4

Routing Table: vpn4
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

B    200.200.202.0/24 [200/0] via 11.11.11.11, 00:39:58
C    200.200.203.0/24 is directly connected, Tunnel7
S    192.168.8.0/24 [1/0] via 200.200.203.2
B    192.168.9.0/24 [200/0] via 11.11.11.11, 00:39:58
     172.16.0.0/24 is subnetted, 2 subnets
S       172.16.8.0 is directly connected, Tunnel7
B       172.16.9.0 [200/0] via 11.11.11.11, 00:39:58
Provider_Edge_1#
```

ANEXO F

Pruebas de Conectividad VPN y Resultados

Anexo F-1.- Conectividad entre VRFs de la VPN1

Ruteador PE0

```
Provider_Edge_0#
Provider_Edge_0#ping vrf vpn1 192.168.8.1 repeat 200

Type escape sequence to abort.
Sending 200, 100-byte ICMP Echos to 192.168.8.1, timeout is 2 seconds:
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
Success rate is 100 percent (200/200), round-trip min/avg/max = 8/55/224 ms
Provider_Edge_0#
```

Ruteador PE1

```
Provider_Edge_1#ping vrf vpn1 192.168.10.1 repeat 200

Type escape sequence to abort.
Sending 200, 100-byte ICMP Echos to 192.168.10.1, timeout is 2 seconds:
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
Success rate is 100 percent (200/200), round-trip min/avg/max = 8/55/156 ms
Provider_Edge_1#
```

Anexo F-2.- Conectividad entre VRFs de la VPN2

Ruteador PE0

```
Provider_Edge_0#ping vrf vpn2 192.168.7.1 repeat 200

Type escape sequence to abort.
Sending 200, 100-byte ICMP Echos to 192.168.7.1, timeout is 2 seconds:
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
Success rate is 100 percent (200/200), round-trip min/avg/max = 8/53/212 ms
Provider_Edge_0#
```

Ruteador PE1

```
Provider_Edge_1#ping vrf vpn2 192.168.9.1 repeat 200

Type escape sequence to abort.
Sending 200, 100-byte ICMP Echos to 192.168.9.1, timeout is 2 seconds:
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
Success rate is 100 percent (200/200), round-trip min/avg/max = 8/57/148 ms
Provider_Edge_1#
```

Anexo F3.- Conectividad entre VRFs de la VPN3

Ruteador PE0

```
Provider_Edge_0#ping vrf vpn3 200.200.201.1 repeat 200

Type escape sequence to abort.
Sending 200, 100-byte ICMP Echos to 200.200.201.1, timeout is 2 seconds:
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
Success rate is 100 percent (200/200), round-trip min/avg/max = 12/39/108 ms
Provider_Edge_0#
```

Ruteador PE1

```
Provider_Edge_1#ping vrf vpn3 200.200.200.1 repeat 200

Type escape sequence to abort.
Sending 200, 100-byte ICMP Echos to 200.200.200.1, timeout is 2 seconds:
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
Success rate is 100 percent (200/200), round-trip min/avg/max = 12/38/108 ms
Provider_Edge_1#
```

Anexo F-4.- Conectividad entre VRFs de la VPN4

Ruteador PE0

```
Provider_Edge_0#ping vrf vpn4 200.200.203.1 repeat 200

Type escape sequence to abort.
Sending 200, 100-byte ICMP Echos to 200.200.203.1, timeout is 2 seconds:
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
Success rate is 100 percent (200/200), round-trip min/avg/max = 12/33/80 ms
Provider_Edge_0#
```

Ruteador PE1

```
Provider_Edge_1#ping vrf vpn4 200.200.202.1 repeat 200

Type escape sequence to abort.
Sending 200, 100-byte ICMP Echos to 200.200.202.1, timeout is 2 seconds:
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
Success rate is 100 percent (200/200), round-trip min/avg/max = 8/40/108 ms
Provider_Edge_1#
```

ANEXO G

Conectividad entre sitios de los Clientes pertenecientes a las VPNs

Anexo G-1.- VPN1

Customer 0_1 (Sitio A)

```
Customer_0_1#ping 192.168.8.1 repeat 200

Type escape sequence to abort.
Sending 200, 100-byte ICMP Echos to 192.168.8.1, timeout is 2 seconds:
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
Success rate is 100 percent (200/200), round-trip min/avg/max = 16/67/132 ms
Customer_0_1#
Customer_0_1#traceroute 192.168.8.1

Type escape sequence to abort.
Tracing the route to 192.168.8.1

  1 172.16.10.1 64 msec 24 msec 20 msec
  2 131.0.0.1 [MPLS: Labels 20/30 Exp 0] 144 msec 92 msec 116 msec
  3 132.0.0.2 [MPLS: Labels 21/30 Exp 0] 64 msec 60 msec 64 msec
  4 172.16.8.1 [MPLS: Label 30 Exp 0] 72 msec 24 msec 52 msec
  5 172.16.8.2 76 msec 72 msec *
Customer_0_1#
```

Customer 1_1 (Sitio B)

```
Customer_1_1#ping 192.168.10.1 repeat 200

Type escape sequence to abort.
Sending 200, 100-byte ICMP Echos to 192.168.10.1, timeout is 2 seconds:
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
Success rate is 100 percent (200/200), round-trip min/avg/max = 28/81/172 ms
Customer_1_1#
Customer_1_1#traceroute 192.168.10.1

Type escape sequence to abort.
Tracing the route to 192.168.10.1

  1 172.16.8.1 4 msec 12 msec 8 msec
  2 133.0.0.1 [MPLS: Labels 22/29 Exp 0] 28 msec 32 msec 32 msec
  3 132.0.0.1 [MPLS: Labels 21/29 Exp 0] 40 msec 44 msec 48 msec
  4 172.16.10.1 [MPLS: Label 29 Exp 0] 48 msec 32 msec 20 msec
  5 172.16.10.2 80 msec 32 msec *
Customer_1_1#
```

Anexo G-2.- VPN2

Customer 0_2 (Sitio A)

```
Customer_0_2#ping 192.168.7.1 repeat 200

Type escape sequence to abort.
Sending 200, 100-byte ICMP Echos to 192.168.7.1, timeout is 2 seconds:
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
Success rate is 100 percent (200/200), round-trip min/avg/max = 16/69/176 ms
Customer_0_2#
Customer_0_2#traceroute 192.168.7.1

Type escape sequence to abort.
Tracing the route to 192.168.7.1

  1 172.16.9.1 8 msec 40 msec 16 msec
  2 131.0.0.1 [MPLS: Labels 20/32 Exp 0] 68 msec 72 msec 104 msec
  3 132.0.0.2 [MPLS: Labels 21/32 Exp 0] 44 msec 64 msec 56 msec
  4 172.16.7.1 [MPLS: Label 32 Exp 0] 60 msec 80 msec 36 msec
  5 172.16.7.2 64 msec 60 msec *
Customer_0_2#
```

Customer 1_2 (Sitio B)

```
Customer_1_1#ping 192.168.9.1 repeat 200

Type escape sequence to abort.
Sending 200, 100-byte ICMP Echos to 192.168.9.1, timeout is 2 seconds:
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
Success rate is 100 percent (200/200), round-trip min/avg/max = 8/68/164 ms
Customer_1_1#
Customer_1_1#traceroute 192.168.9.1

Type escape sequence to abort.
Tracing the route to 192.168.9.1

  1 200.200.203.1 12 msec 12 msec 20 msec
  2 133.0.0.1 [MPLS: Labels 22/36 Exp 0] 60 msec 28 msec 60 msec
  3 132.0.0.1 [MPLS: Labels 21/36 Exp 0] 40 msec 36 msec 52 msec
  4 200.200.202.1 [MPLS: Label 36 Exp 0] 68 msec 36 msec 40 msec
  5 200.200.202.2 36 msec 12 msec *
Customer_1_1#
```

Anexo G-3.- VPN3

Customer 0_1 (Sitio A)

```
Customer_0_1#ping 192.168.7.1 repeat 200

Type escape sequence to abort.
Sending 200, 100-byte ICMP Echos to 192.168.7.1, timeout is 2 seconds:
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
Success rate is 100 percent (200/200), round-trip min/avg/max = 20/76/156 ms
Customer_0_1#
Customer_0_1#traceroute 192.168.7.1

Type escape sequence to abort.
Tracing the route to 192.168.7.1

  1 200.200.200.1 44 msec 32 msec 16 msec
  2 131.0.0.1 [MPLS: Labels 20/34 Exp 0] 60 msec 124 msec 88 msec
  3 132.0.0.2 [MPLS: Labels 21/34 Exp 0] 68 msec 76 msec 20 msec
  4 200.200.201.1 [MPLS: Label 34 Exp 0] 60 msec 52 msec 96 msec
  5 200.200.201.2 64 msec 60 msec *
Customer_0_1#
```

Customer 1_2 (Sitio B)

```
Customer_1_2#ping 192.168.10.1 repeat 200

Type escape sequence to abort.
Sending 200, 100-byte ICMP Echos to 192.168.10.1, timeout is 2 seconds:
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
Success rate is 100 percent (200/200), round-trip min/avg/max = 8/65/168 ms
Customer 1 2#
Customer_1_2#traceroute 192.168.10.1

Type escape sequence to abort.
Tracing the route to 192.168.10.1

  1 200.200.201.1 24 msec 16 msec 24 msec
  2 133.0.0.1 [MPLS: Labels 22/33 Exp 0] 32 msec 68 msec 44 msec
  3 132.0.0.1 [MPLS: Labels 21/33 Exp 0] 52 msec 36 msec 44 msec
  4 200.200.200.1 [MPLS: Label 33 Exp 0] 36 msec 24 msec 32 msec
  5 200.200.200.2 40 msec 24 msec *
Customer_1_2#
```

Anexo G-4.- VPN4

Customer 0_2 (Sitio A)

```
Customer_0_2#ping 192.168.8.1 repeat 200

Type escape sequence to abort.
Sending 200, 100-byte ICMP Echos to 192.168.8.1, timeout is 2 seconds:
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
Success rate is 100 percent (200/200), round-trip min/avg/max = 16/68/156 ms
Customer_0_2#
Customer_0_2#traceroute 192.168.8.1

Type escape sequence to abort.
Tracing the route to 192.168.8.1

  1 200.200.202.1 40 msec 36 msec 12 msec
  2 131.0.0.1 [MPLS: Labels 20/37 Exp 0] 68 msec 64 msec 76 msec
  3 132.0.0.2 [MPLS: Labels 21/37 Exp 0] 32 msec 84 msec 72 msec
  4 200.200.203.1 [MPLS: Label 37 Exp 0] 20 msec 100 msec 16 msec
  5 200.200.203.2 108 msec 76 msec *
Customer_0_2#
```

Customer 1_1 (Sitio B)

```
Customer_1_1#ping 192.168.9.1 repeat 200

Type escape sequence to abort.
Sending 200, 100-byte ICMP Echos to 192.168.9.1, timeout is 2 seconds:
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
Success rate is 100 percent (200/200), round-trip min/avg/max = 8/68/164 ms
Customer_1_1#
Customer_1_1#traceroute 192.168.9.1

Type escape sequence to abort.
Tracing the route to 192.168.9.1

  1 200.200.203.1 12 msec 12 msec 20 msec
  2 133.0.0.1 [MPLS: Labels 22/36 Exp 0] 60 msec 28 msec 60 msec
  3 132.0.0.1 [MPLS: Labels 21/36 Exp 0] 40 msec 36 msec 52 msec
  4 200.200.202.1 [MPLS: Label 36 Exp 0] 68 msec 36 msec 40 msec
  5 200.200.202.2 36 msec 12 msec *
Customer 1 1#
```

ANEXO H

Prueba de Actividad de los Túneles de Ingeniería de Tráfico en los dispositivos de Frontera

Anexo H-1.- Provider Edge 0 (PE0)

```
Provider_Edge_0#show mpls traffic-eng tunnels tunnel 1

Name: Provider_Edge_0_t1                 (Tunnel1) Destination: 17.17.17.17
  Status:
    Admin: up         Oper: up     Path: valid       Signalling: connected

    path option 1, type dynamic (Basis for Setup, path weight 36)

  Config Parameters:
    Bandwidth: 2048     kbps (Global)  Priority: 1  1   Affinity: 0x0/0xFFFF
    Metric Type: TE (default)
    AutoRoute:  enabled   LockDown: disabled  Loadshare: 2048     bw-based
    auto-bw: disabled

  InLabel  :  -
  OutLabel : FastEthernet0/0, 20
  RSVP Signalling Info:
       Src 11.11.11.11, Dst 17.17.17.17, Tun_Id 1, Tun_Instance 9
    RSVP Path Info:
      My Address: 131.0.0.2
      Explicit Route: 131.0.0.1 132.0.0.2 133.0.0.2 17.17.17.17
      Record Route:  NONE
      Tspec: ave rate=2048 kbits, burst=1000 bytes, peak rate=2048 kbits
    RSVP Resv Info:
      Record Route:  NONE
      Fspec: ave rate=2048 kbits, burst=1000 bytes, peak rate=2048 kbits
  Shortest Unconstrained Path Info:
    Path Weight: 36 (TE)
    Explicit Route: 131.0.0.2 131.0.0.1 132.0.0.2 133.0.0.2
                    17.17.17.17
  History:
    Tunnel:
      Time since created: 1 hours, 42 minutes
      Time since path change: 1 hours, 40 minutes
    Current LSP:
      Uptime: 1 hours, 40 minutes
Provider_Edge_0#
```

Anexo H-2.- Provider Edge 1 (PE1)

```
Provider_Edge_1#show mpls traffic-eng tunnels tunnel 4

Name: Provider_Edge_1_t4                  (Tunnel4) Destination: 11.11.11.11
  Status:
    Admin: up         Oper: up     Path: valid       Signalling: connected

    path option 1, type dynamic (Basis for Setup, path weight 36)

  Config Parameters:
    Bandwidth: 2048     kbps (Global)  Priority: 1  1   Affinity: 0x0/0xFFFF
    Metric Type: TE (default)
    AutoRoute:  enabled   LockDown: disabled  Loadshare: 2048     bw-based
    auto-bw: disabled

  InLabel  :  -
  OutLabel : Serial1/3, 22
  RSVP Signalling Info:
       Src 17.17.17.17, Dst 11.11.11.11, Tun_Id 4, Tun_Instance 10
    RSVP Path Info:
      My Address: 17.17.17.17
      Explicit Route: 133.0.0.1 132.0.0.1 131.0.0.1 131.0.0.2
                      11.11.11.11
      Record Route:  NONE
      Tspec: ave rate=2048 kbits, burst=1000 bytes, peak rate=2048 kbits
    RSVP Resv Info:
      Record Route:  NONE
      Fspec: ave rate=2048 kbits, burst=1000 bytes, peak rate=2048 kbits
  Shortest Unconstrained Path Info:
    Path Weight: 36 (TE)
    Explicit Route: 133.0.0.1 132.0.0.1 131.0.0.1 131.0.0.2
                    11.11.11.11
  History:
    Tunnel:
      Time since created: 1 hours, 35 minutes
      Time since path change: 1 hours, 35 minutes
    Current LSP:
      Uptime: 1 hours, 35 minutes
Provider_Edge_1#
```

ANEXO I

Calidad de Servicio (QoS) Paquetes Clasificados y Marcados viajando a través del Backbone

Anexo I-1.- Mapas de Políticas en los Dispositivos de Frontera

Provider Edge 0 (PE0)

```
Provider_Edge_0#show policy-map
  Policy Map output_map
    Class video_in
      Bandwidth 20 (%) Max Threshold 64 (packets)
    Class voice_in
      Strict Priority
      Bandwidth 25 (%)
    Class ftp_in
      Bandwidth 6 (%) Max Threshold 64 (packets)
    Class mail_in
      Bandwidth 12 (%) Max Threshold 64 (packets)
    Class telnet_in
      Bandwidth 8 (%) Max Threshold 64 (packets)
  Policy Map input_map
    Class voice_in
      set ip precedence 5
     police cir 128000 bc 4000 be 4000
       conform-action transmit
       exceed-action transmit
       violate-action transmit
    Class video_in
      set ip precedence 4
     police cir 128000 bc 4000 be 4000
       conform-action transmit
       exceed-action transmit
       violate-action transmit
    Class mail_in
      set ip precedence 3
     police cir 64000 bc 2000 be 2000
       conform-action transmit
       exceed-action set-prec-transmit 0
       violate-action drop
    Class telnet_in
      set ip precedence 2
     police cir 64000 bc 2000 be 2000
       conform-action transmit
       exceed-action set-prec-transmit 0
       violate-action drop
    Class ftp_in
      set ip precedence 1
     police cir 64000 bc 2000 be 2000
       conform-action transmit
       exceed-action set-prec-transmit 0
       violate-action drop
    Class class-default
     police cir 64000 bc 2000 be 2000
       conform-action transmit
       exceed-action transmit
       violate-action drop
Provider_Edge_0#
```

Provider Edge 1 (PE1)

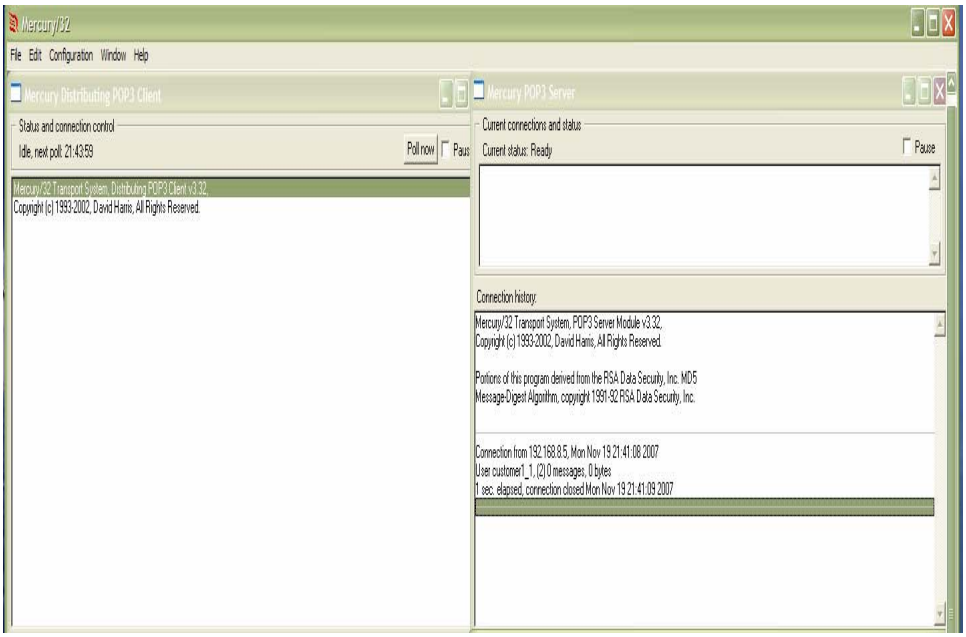
```
Provider_Edge_1#show policy-map input_map
  Policy Map input_map
    Class voice_in
      set ip precedence 5
     police cir 128000 bc 4000 be 4000
       conform-action transmit
       exceed-action transmit
       violate-action transmit
    Class video_in
      set ip precedence 4
     police cir 128000 bc 4000 be 4000
       conform-action transmit
       exceed-action transmit
       violate-action transmit
    Class mail_in
      set ip precedence 3
     police cir 64000 bc 2000 be 2000
       conform-action transmit
       exceed-action set-prec-transmit 0
       violate-action drop
    Class telnet_in
      set ip precedence 2
     police cir 64000 bc 2000 be 2000
       conform-action transmit
       exceed-action set-prec-transmit 0
       violate-action drop
    Class ftp_in
      set ip precedence 1
     police cir 64000 bc 2000 be 2000
       conform-action transmit
       exceed-action set-prec-transmit 0
       violate-action drop
    Class class-default
     police cir 64000 bc 2000 be 2000
       conform-action transmit
       exceed-action transmit
       violate-action drop
Provider_Edge_1#show policy-map output_map
  Policy Map output_map
    Class video_in
      Bandwidth 20 (%) Max Threshold 64 (packets)
    Class voice_in
      Strict Priority
      Bandwidth 25 (%)
    Class ftp_in
      Bandwidth 6 (%) Max Threshold 64 (packets)
    Class mail_in
      Bandwidth 12 (%) Max Threshold 64 (packets)
    Class telnet_in
      Bandwidth 8 (%) Max Threshold 64 (packets)
Provider_Edge_1#
```

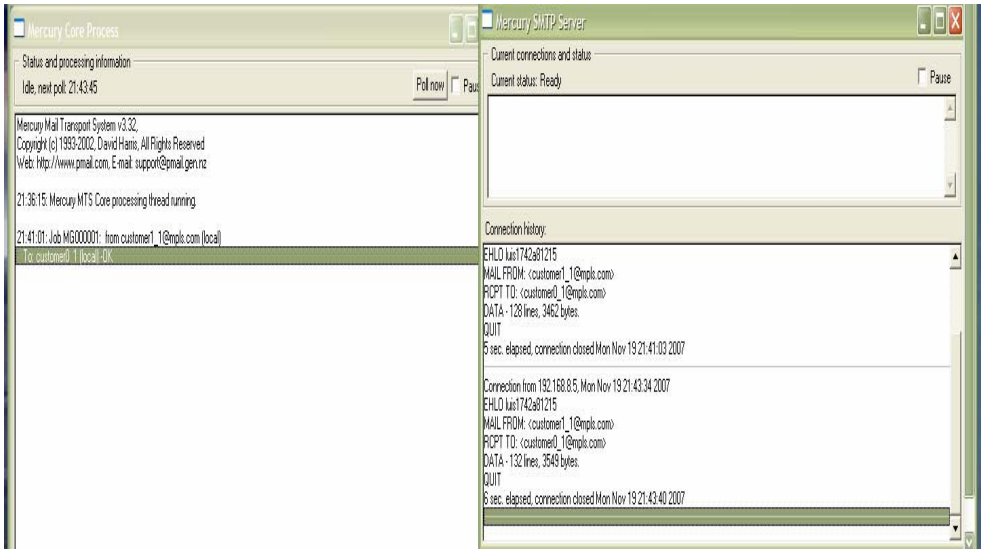
ANEXO J

Resultados

Anexo J-1

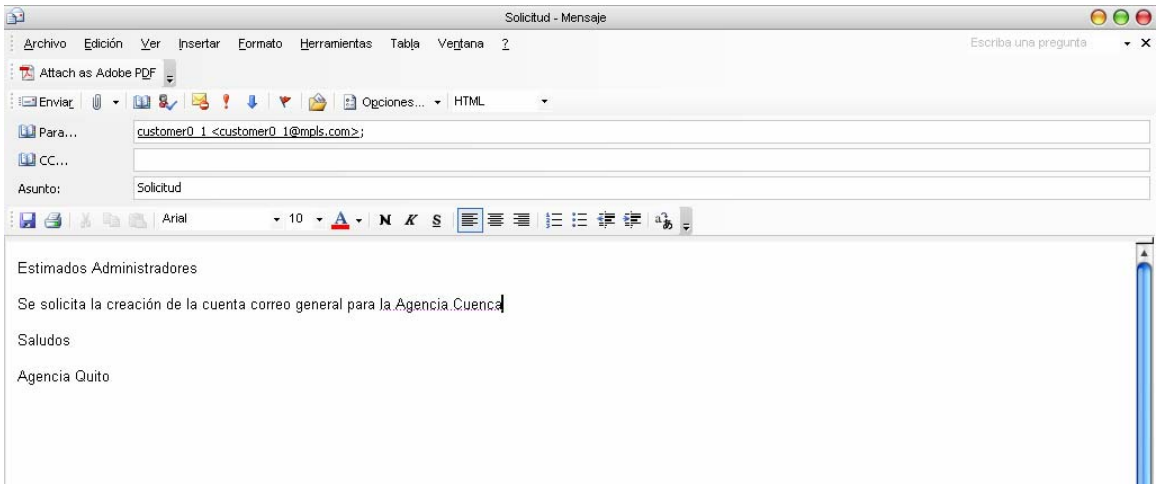
Anexo J-1-1.- Pruebas de Aplicaciones de Correo Electrónico (SMTP) – Servidor "MERCURY"



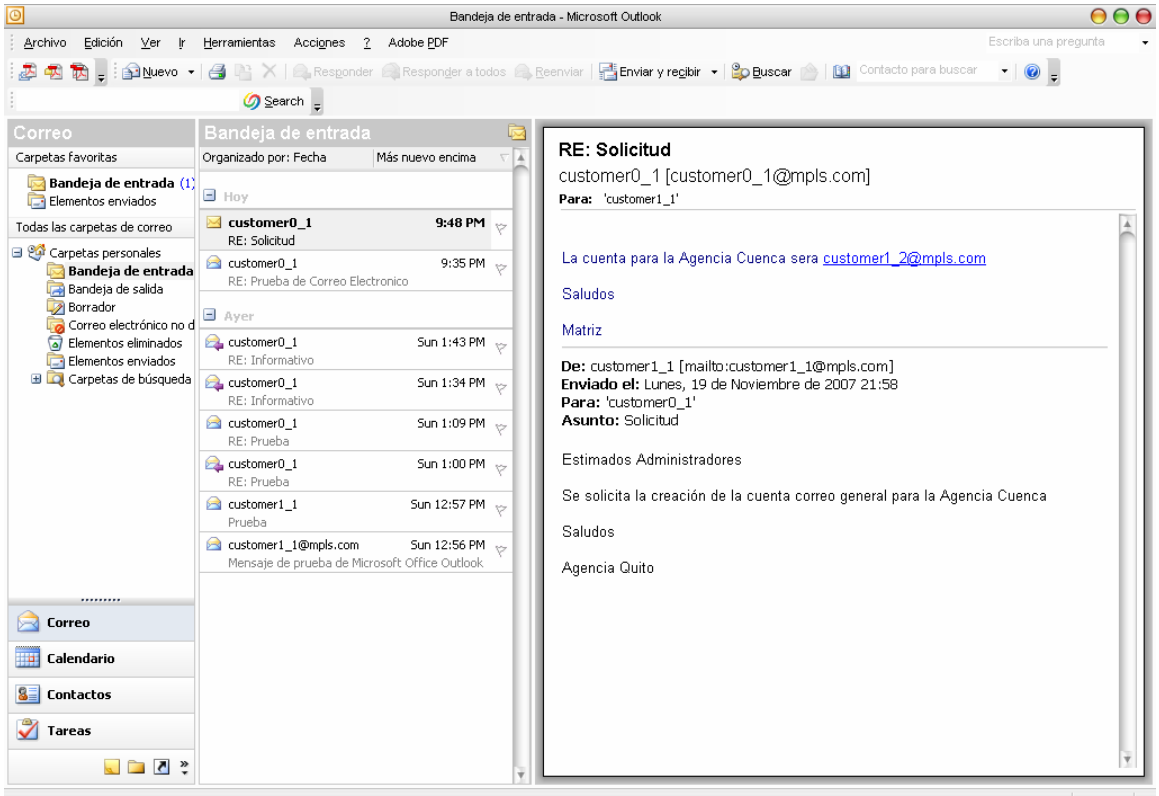


Anexo J-1-2.- Envío de Correo Electrónico con el uso de Outlook

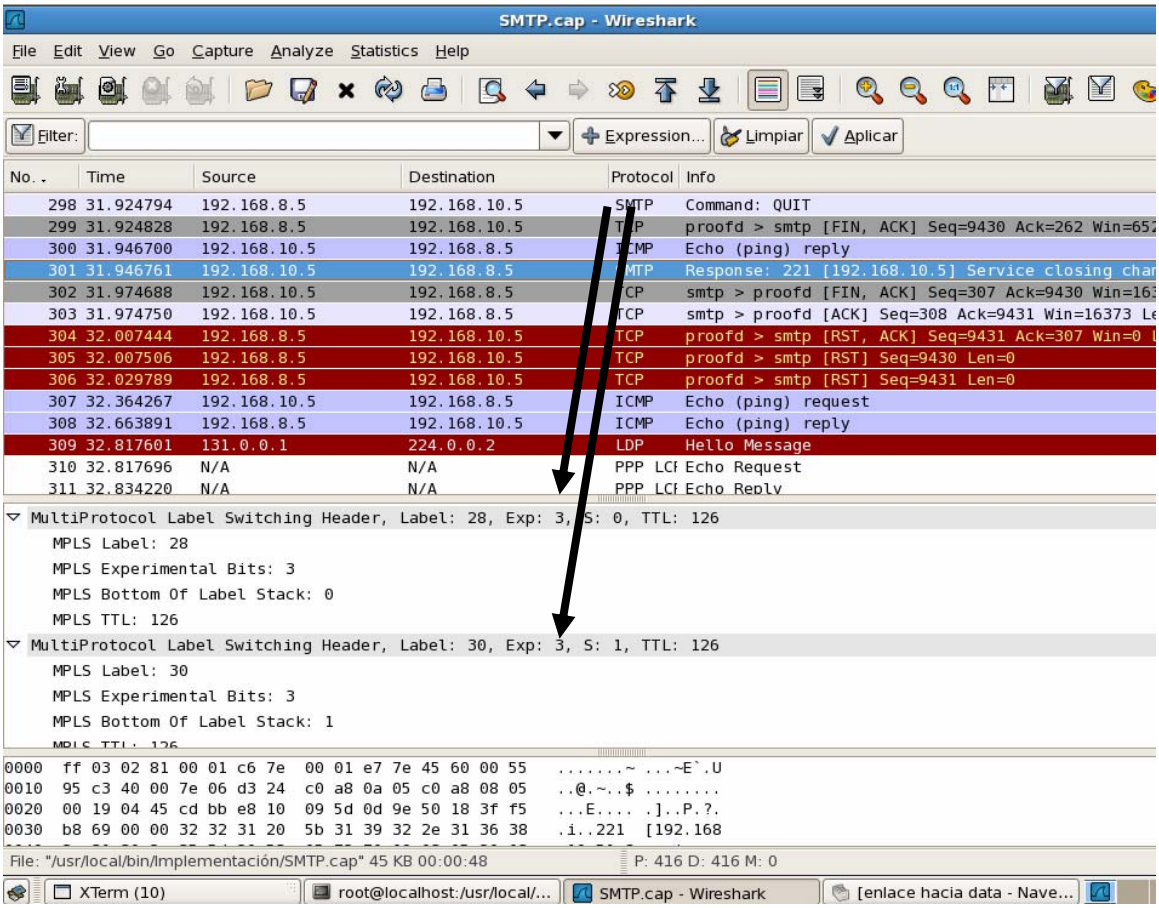
Envío de Correo desde Agencia Quito a Matriz Guayaquil



Recepción de Correo en la Matriz Guayaquil

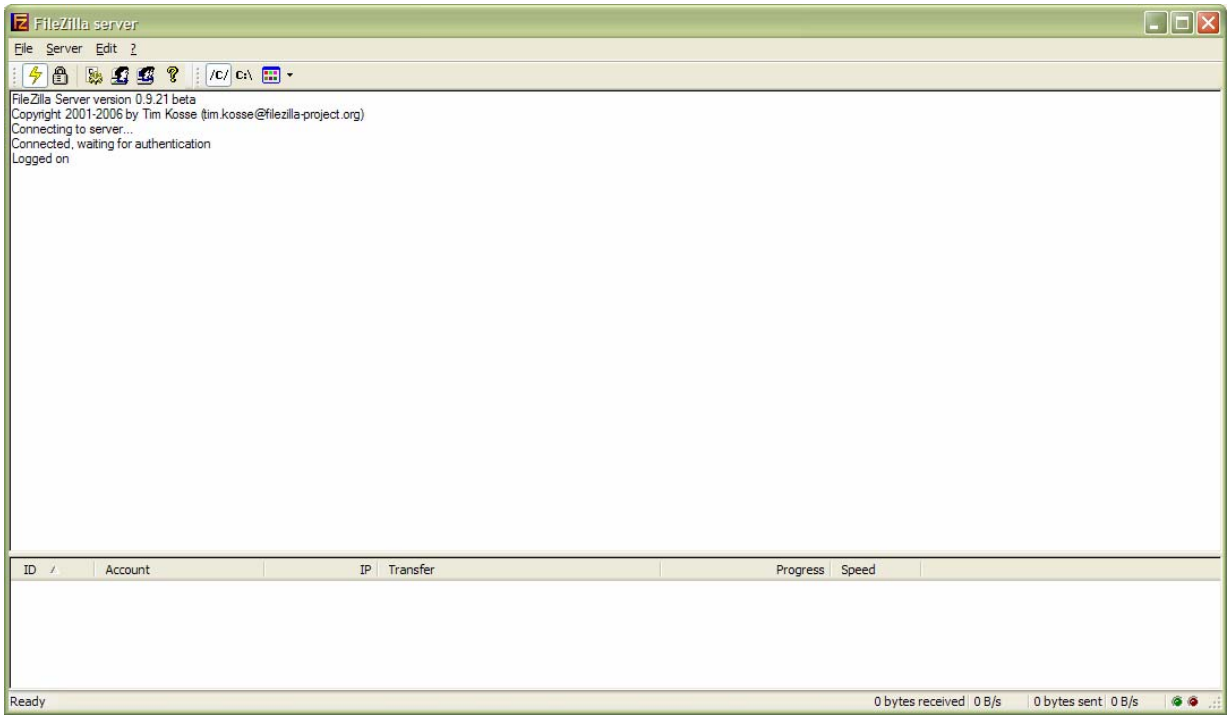


Anexo J-1-3.- Viaje y Análisis del Paquete SMTP Marcado con prioridad de Calidad de Servicio

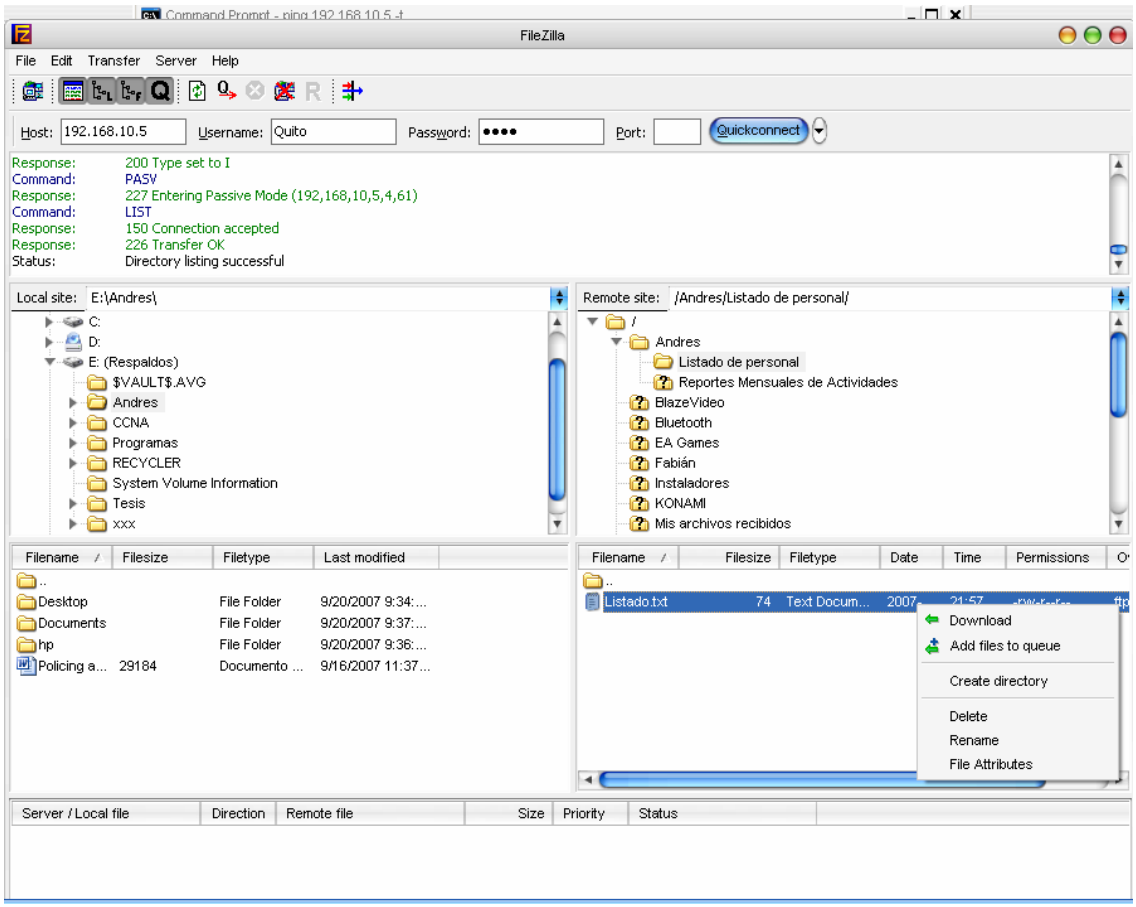


Anexo J-2

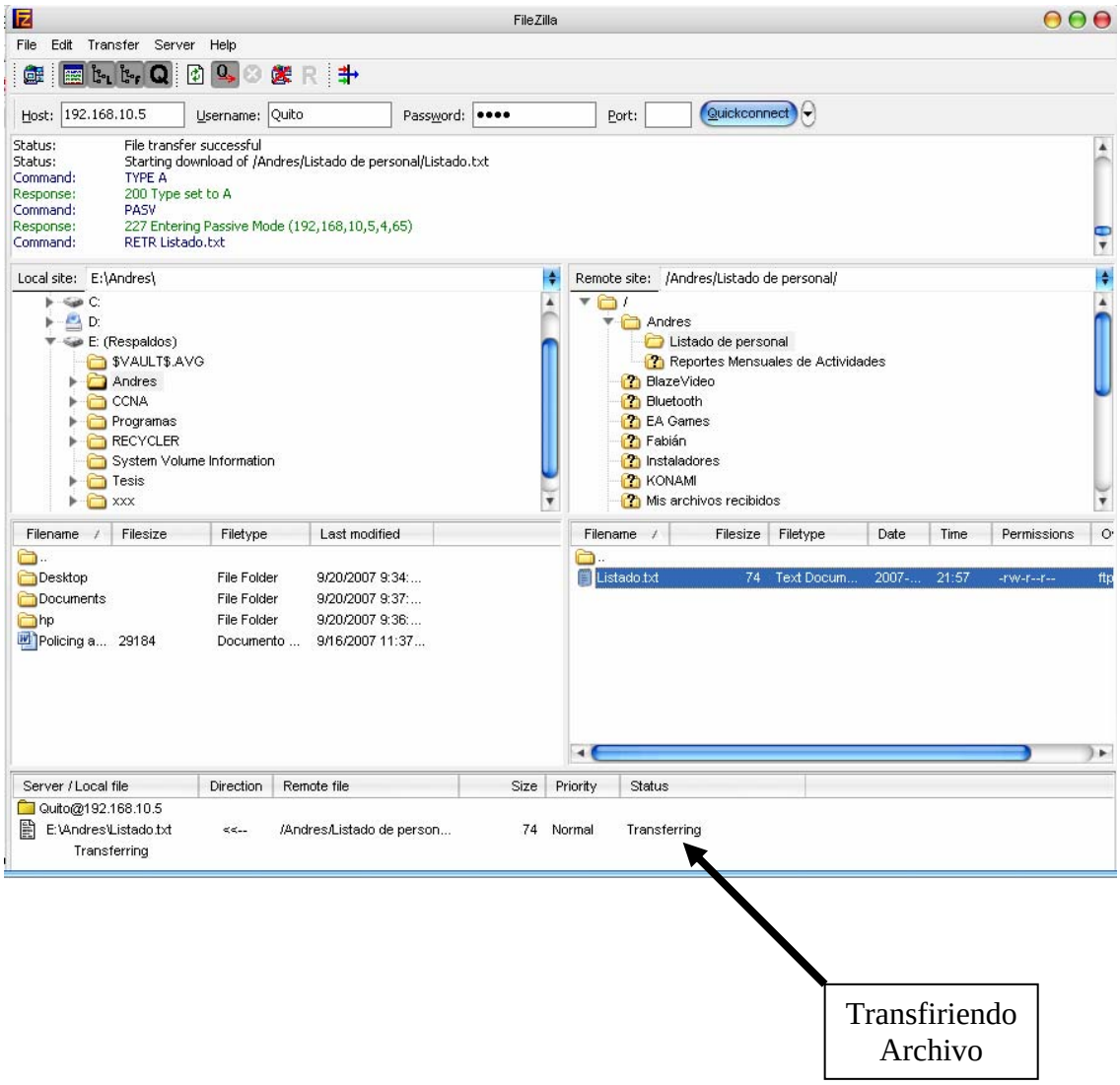
Anexo J-2-1.-Pruebas de Aplicaciones de transferencia de Archivos (FTP) – Servidor “FILEZILLA”



Anexo J-2-2.- Utilización de FILEZILLA CLIENT en Agencia Quito para descarga de Archivos desde un Servidor ubicado en la Matriz

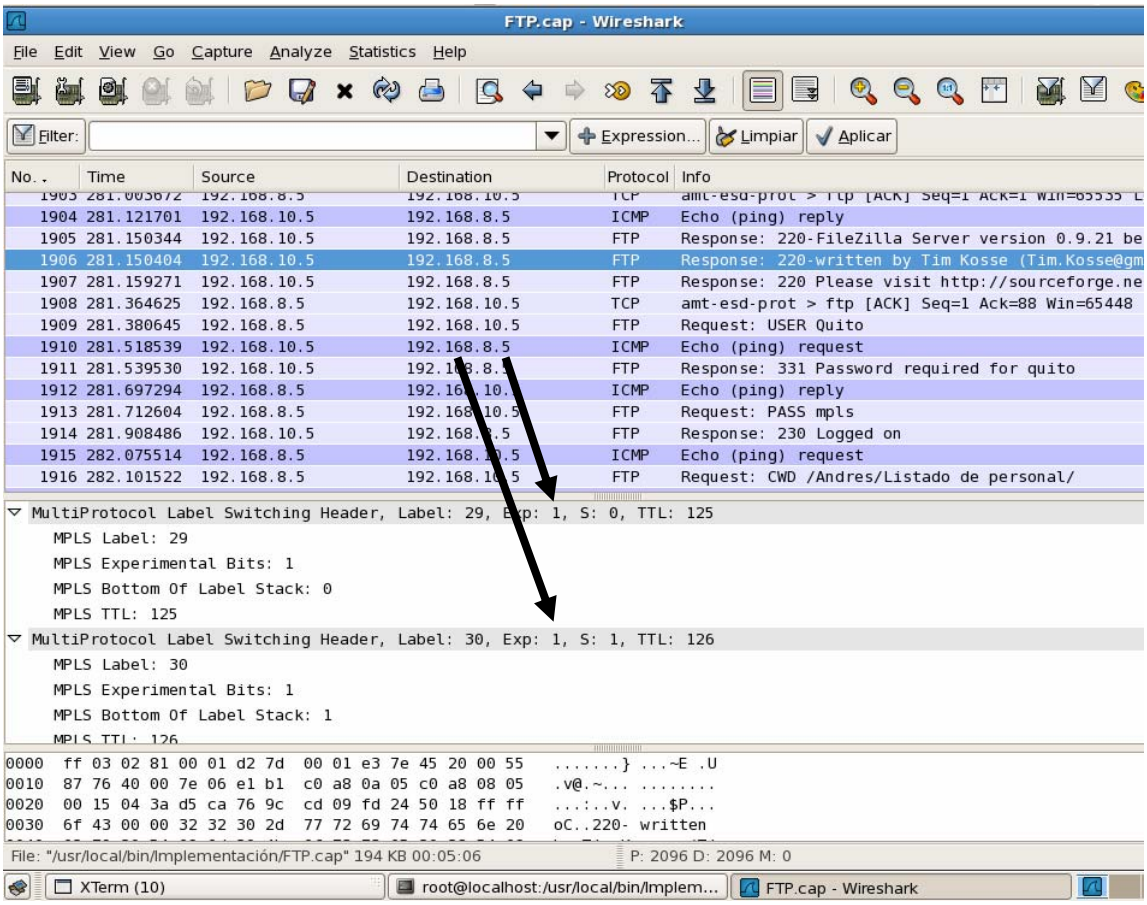


Anexo J-2-3.- Inicio de la Transferencia de Archivos



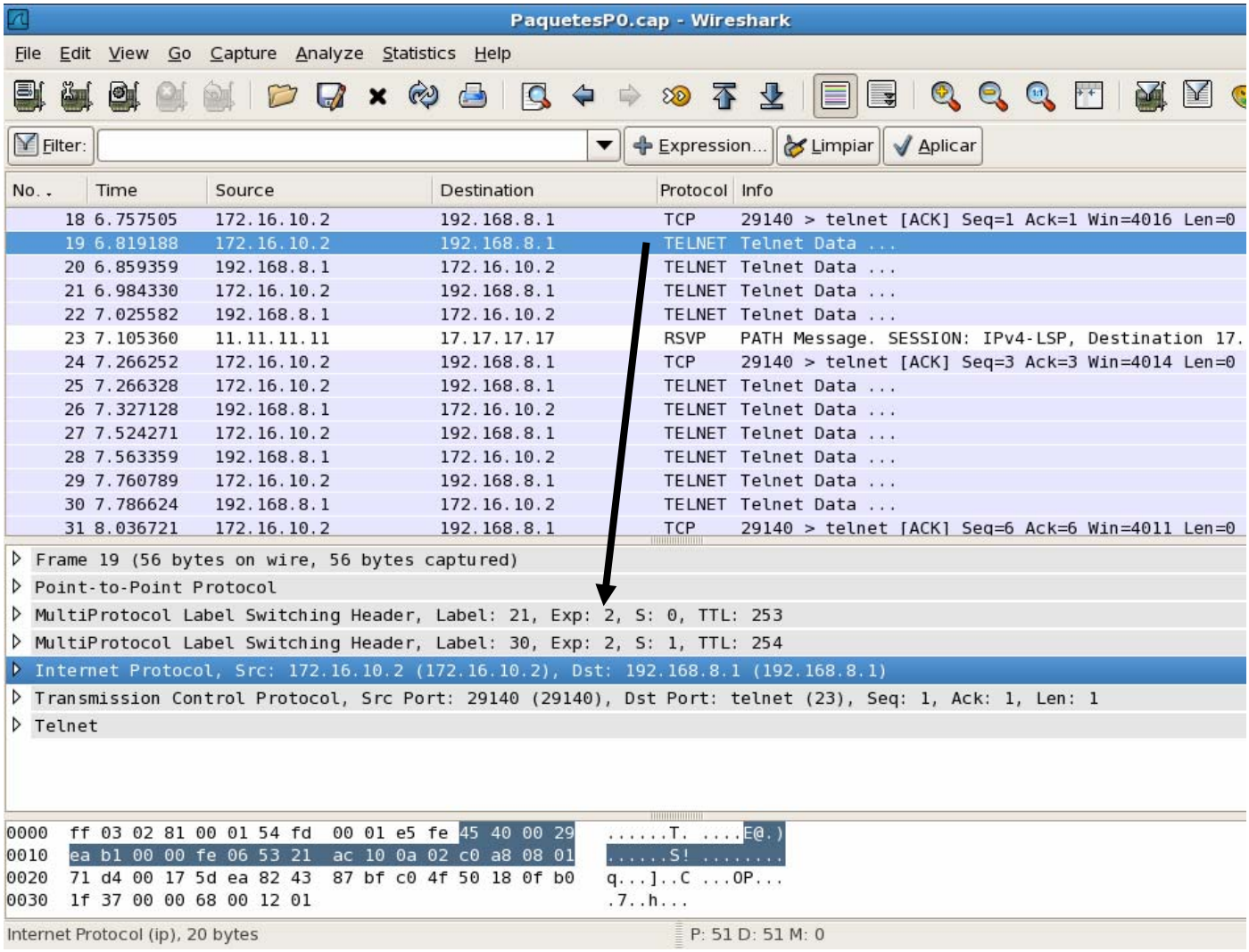
Transfiriendo Archivo

Anexo J-2-4.-Viaje y Análisis del Paquete FTP Marcado con prioridad de Calidad de Servicio



Anexo J-3

Anexo J-3-1.- Captura de Paquetes en la Interfaz Serial ½ del Router Provider_0 (P0). Clase “TELNET”. Paquetes Marcados con Prioridad 2



Anexo J-3-2.- Captura de Paquetes en la Interfaz Serial ½ del Router Provider_1 (P1). Clase TELNET. Paquetes Marcados con Prioridad 2.