Universidad Politécnica de Madrid

Escuela Técnica Superior de Ingenieros de Telecomunicación

ESTUDIO DE LA INGENIERÍA DE TRÁFICO EN REDES MPLS MEDIANTE CASOS DE USO PRÁCTICO CON LA HERRAMIENTA VNX

TRABAJO FIN DE MÁSTER

Tatiana Hernández Camacho

2015

Universidad Politécnica de Madrid

Escuela Técnica Superior de Ingenieros de Telecomunicación

Máster Universitario en
Ingeniería de Redes y Servicios Telemáticos

TRABAJO FIN DE MÁSTER

ESTUDIO DE LA INGENIERÍA DE TRÁFICO EN REDES MPLS MEDIANTE CASOS DE USO PRÁCTICO CON LA HERRAMIENTA VNX

Autor

Tatiana Hernández Camacho

Director

David Fernández Cambronero

Departamento de Ingeniería de Sistemas Telemáticos

2015

Resumen

El volumen de tráfico en Internet está creciendo de manera exponencial. Este crecimiento acelerado de los datos supone un reto para los operadores de servicio que deben implementar mecanismos que les permitan enfrentar a la creciente demanda. El algoritmo de enrutamiento utilizado por los IGPs es el de "la ruta más corta", el resultado es que ciertos enlaces en la red se ven congestionados, mientras existen otras rutas disponibles que no son utilizadas. Este tipo de enrutamiento provoca demoras impredecibles y pérdida de datos. Sin embargo, no ha sido un problema para las aplicaciones tradicionales de Internet como web, correo electrónico, transferencia de archivos y similares, pero la nueva generación de aplicaciones que incluyen audio y video streaming, exigen alto rendimiento ancho de banda y baja latencia.

Por tanto, es un objetivo clave para los operadores de servicio gestionar el tráfico y recursos de red de la manera más eficiente posible, utilizado técnicas que permitan adaptar los distintos flujos de datos a los recursos de ancho de banda y hardware disponibles en el núcleo de red, de manera que obtengan las garantías necesarias para el correcto funcionamiento de las aplicaciones.

La Ingeniería de Trafico ofrece varios mecanismos para optimizar el rendimiento, modelado, medición, caracterización y el control de tráfico en una red para obtener objetivos específicos de rendimiento y ofrecer servicios competitivos a los clientes. MPLS constituye un elemento clave en el despliegue de técnicas de Ingeniería de Trafico ya que la idea básica de MPLS es separar completamente el plano de control (enrutamiento) del plano de datos (reenvío de paquetes) mientras mantiene la compatibilidad con las infraestructuras de red IP existentes.

Para lograr esto MPLS coloca una cabecera adicional a los paquetes IP que contienen la información necesaria para la toma de decisiones de envío. En consecuencia, en los routers habilitados con MPLS el tráfico no se reenvía basado en el algoritmos de la "ruta más corta" sino que los paquetes son agrupados por clases o FEC (Forwarding Equivalence Class) en el momento en que entran en el dominio MPLS y basado en esta clasificación son asignados a un LSP (Label Switched Path) y reenviados al router de salida. De esta manera el administrador de red tiene un control total sobre la clasificación de paquetes y el establecimiento de rutas.

En esta memoria se realiza una revisión del estado del arte de la tecnología y la implementación de cuatro casos prácticos utilizando Ingeniería de Trafico en redes

MPLS, con varios escenarios virtuales emulados utilizando la herramienta de simulación de redes VNX. En cada uno de los casos realizados: (1) Enrutamiento de tráfico mediante el uso de túneles TE, (2) Uso del mecanismo de Fast Reroute, (3) Implementación de balanceo de carga en enlaces de diferente costo y (4) Calidad de servicio en MPLS-TE, se realizaron mediciones de prestaciones como el rendimiento, jitter y perdida de datagramas. Estas mediciones se realizaron en una red MPLS que utiliza Ingeniería de Tráfico y una red MPLS puro, de manera que mediante la comparación y análisis cuantitativo de los resultados obtenidos se llegue a un mejor conocimiento y comprensión de los beneficios y ventajas que ofrece MPLS-TE con respecto a otras arquitecturas de red.

Abstract

The Internet traffic volume is growing exponentially. This accelerated data growth represents a challenge to service providers who must implement mechanisms that allow them to cope with the growing demand. The routing algorithm used by the IGP is the "shortest path first", the result is that IP provides a service called best effort. This type of service is subject to unpredictable delays and data loss. The best effort model has not been a problem for traditional Internet applications such as web, email, file transfer and the like. But, the new generation applications including streaming audio and video, require high-performance with wide bandwidth and low latency.

Therefore, it is a key objective for service operators to manage traffic and network resources as efficiently as possible, using techniques to match different data streams to bandwidth resources and hardware available in the backbone they obtain the guarantees necessary for the proper operation of applications.

Traffic Engineering provides several mechanisms to optimize performance, modeling, measurement, characterization and control of network traffic for specific performance targets and offer competitive services to customers. MPLS is a key element in the deployment of Traffic Engineering techniques. The basic idea of MPLS is completely separate control plane (routing) from data plane (packet forwarding), while maintaining compatibility with existing IP network infrastructures.

To accomplish this, MPLS places an additional header to IP packets that contain the information necessary for making forwarding decisions. Consequently, MPLS enabled routers do not forward traffic only based on the algorithms of the "shortest route" but the packages are grouped by classes or FECs (Forwarding Equivalence Class) at the time they enter the MPLS domain, and based on this classification they are assigned to an LSP (Label Switched Path) and forwarded to the egress router. In this way the network administrator has full control over packet classification and route establishment.

This report reviews the state of the art of MPLS Traffic Engineering technology and show the implementation of four cases of study with several virtual environments emulated using VNX network simulation tool. In each of the cases of study: (1) Routing of traffic by using TE tunnels, (2) Using the Fast Reroute mechanism, (3) Implementation of load balancing links different cost and (4) Quality MPLS-TE service, in each of this sceneries has been taken network performance measurement, jitter

measurement and loss of datagrams measurement. These measurements were performed on a network using MPLS Traffic Engineering and a MPLS only network. These results were compared and based on a quantitative analysis of the results, was possible to reach a better knowledge and understanding of the benefits and advantages of MPLS-TE with respect to other network architectures.

Índice general

Índice de figuras

Índice de Tablas

Siglas

IGP	Interior Gateway Protocol
MPLS	Multiprotocol Label Switching
FEC	Forwarding Equivalence Class
LSP	Label Switched Path
MPLS-TE	Multiprotocol Label Switching Traffic Engineering
OSPF	Open Shortest Path First
ISIS	Intermediate System to Intermediate System
ISP	Internet Service Provider
VNX	Virtual Networks over linuX
IETF	Internet Engineering Task Force
QoS	Quality of Service
ATM	Asynchronous Transfer Mode
LSR	Label Switched Router
LER	Label Edge Router
RFC	Request for Comments
VPN	Virtual Private Networks
LDP	Label Distribution Protocol
RSVP	Resource Reservation Protocol
RSVP–TE	Resource Reservation Protocol-Traffic Engineering
CR-LDP	Constraint-based Routing Label Distribution Protocol
BGP	Border Gateway Protocol
LIB	Label Information Base
LFIB	Label Forwarding Information Base
SPF	Shortest Path First
CSPF	Constrained Shortest Path First

1 Introducción

1.1 Contexto

El rápido crecimiento de la Internet ha tenido un gran impacto sobre los tipos de servicios solicitados por los consumidores y el tipo de rendimiento que exigen de los servicios que desean utilizar. En consecuencia a lo anterior, los proveedores de servicios se han visto en la obligación de desarrollar, gestionar y mejorar la infraestructura de red IP en términos de rendimiento y el control del tráfico a través de la ingeniería de tráfico.

Las redes IP a menudo funcionan con los conocidos "*shortest path routing protocols*", que como su nombre lo indica son protocolos de enrutamiento basados en el reenvío de paquetes utilizando la ruta más corta, como por ejemplo OSPF o ISIS. En un momento, cuando varios paquetes originados a partir de diferentes redes comienzan a utilizar el mismo camino o ruta más corta, este camino puede llegar a ser muy cargado y esto dará lugar a la congestión de la red.

Varias técnicas han sido desarrolladas para hacer frente a las deficiencias de los protocolos de enrutamiento basados en la ruta más corta, entre ellas se encuentra la ingeniería de tráfico. Con la ingeniería de tráfico se puede enrutar el tráfico a través de otros caminos, y no únicamente utilizar el camino más corto y el propósito principal de la ingeniería de tráfico es lograr el mayor rendimiento en las redes IP, alta calidad de servicio, eficiencia, y la más alta utilización posible de los recursos de red.

En redes IP tradicionales las herramientas para la ingeniería de tráfico son limitadas debido a la falta de control sobre las rutas utilizadas. Uno de los propósitos de *Multi Protocol Label Switching* (MPLS) es proporcionar más capacidades a TE (*Traffic Engineering*). El enrutamiento explícito permite que los paquetes sean encaminados a través de rutas predefinidas. A medida que el tráfico se divide en múltiples caminos paralelos se obtiene una mejor distribución del tráfico en la red y mejor aprovechamiento de los recursos de red disponibles.

Además de mejorar el rendimiento y utilización de los recursos de red, MPLS-TE usado en conjunto con DiffServ permite ofrecer calidad de servicio de punta a punta, que en redes IP resulta muy difícil de conseguir. Por tanto, es una herramienta muy efectiva para cumplir con los exigentes requerimientos de calidad de servicio de las aplicaciones actuales como son las aplicaciones en tiempo real, voz sobre IP, video, etc.

1.2 Motivación

En la actualidad existe una tendencia entre los ISPs a adoptar MPLS como tecnología base para el núcleo de red, y el uso de MPLS da lugar también al uso de técnicas como la Ingeniería de Trafico, que mediante mecanismos de enrutamiento óptimo como el enrutamiento explicito, permiten a los ISPs coordinar y llevar un mejor control del tráfico que fluye a través de su red y determinar si poseen la capacidad necesaria para tolerar la llegada de nuevas demandas sin afectar las ya existentes.

Existen además situaciones en las que el crecimiento interno del núcleo de red en un ISP se da de manera muy acelerada y en ambientes de poco control, donde las funciones de gestión de red que antes podían ser realizadas de forma manual por un grupo reducido de personas ya resulta insuficiente y limitado. Lo anterior da lugar a la necesidad del uso de mecanismos de Ingeniería de Trafico que permitan manejar los flujos de datos y recursos físicos de la red de manera más eficiente.

Por lo expuesto anteriormente, y como profesional del área de redes, me vi muy interesada en realizar un estudio de la Ingeniería de Trafico y conocer las herramientas que facilita la Ingeniería de Trafico para resolver problemas de congestión de red en ambientes Intra-dominio.

1.3 Objetivos

Los objetivos de esta memoria son los siguientes:

1. Realizar un estudio teórico de la Ingeniería de tráfico en redes MPLS y los medios que utiliza para asegurar el uso eficiente de los recursos de la red en el enrutamiento de tráfico y aseguramiento de la calidad de servicio. Esta base teórica servirá para el desarrollo de la parte práctica del presente trabajo.

2. Realizar el estudio práctico de la Ingeniería de Trafico en redes MPLS intradominio, mediante la implementación de cuatro casos de uso que permitan comprender la tecnología y conocer las principales funcionalidades que se pueden realizar utilizando Ingeniería de Trafico en redes MPLS. Realizando un análisis de las prestaciones y ventajas que se pueden alcanzar con MPLS-TE en comparación con otros tipos de redes. La implementación de estos casos de estudio se realizara mediante el despliegue de un escenario de red virtual utilizando la herramienta de simulación de redes VNX.

2 Multi-Protocol Label Switching (MPLS)

MPLS (Multi-Protocol Label Switching) es una tecnología de conmutación de paquetes de capa 3 que realiza enrutamiento de tráfico de manera efectiva, facilita el despliegue de técnicas de QoS y está estandarizada por la Internet Engineering Task Force (IETF) [48]. MPLS es capaz de establecer conexiones IP punto-a-punto con diferentes requisitos de calidad de servicio asociadas a varios tipos de flujos permitiendo un mejor y eficiente enrutamiento en la red [14], [43].

MPLS utiliza la técnica de conmutación de etiquetas [47] que le proporciona la posibilidad de administrar el tráfico de una red a través de etiquetas en las cabeceras de los paquetes y a routers específicos capaces de reconocerlas. Principalmente consiste en integrar los niveles de enlace y red eficazmente. Es decir, combina la inteligencia del routing con la velocidad del switching.

MPLS usa un esquema de etiquetado de tráfico, marcándolo en la entrada de la red, pero no en su salida. Es usado únicamente en los routers y es independiente del protocolo usado, lo que le permite ser utilizado sobre otros protocolos distintos a IP. Además soporta varias tecnologías de acceso que incluyen T1/E1, ATM, Frame Relay, DSL. Los protocolos de enrutamiento de nivel 3 como OSPF o IS-IS se usan únicamente para funciones de control, ya que las decisiones de enrutamiento se toman en función de la etiqueta MPLS y no de la cabecera IP.

MPLS mejora la escalabilidad de la red, reduciendo las tablas de enrutamiento y el retardo de proceso en los routers [37], combinando algunas prestaciones de las redes orientadas a conexión con la de las redes sin conexión. Así, un router asigna una etiqueta a cada una de las entradas de la tabla de enrutamiento y las distribuye a sus routers vecinos. Luego, cuando se pasan paquetes entre ellos, los routers solo tienen que leer la etiqueta MPLS para identificar el siguiente salto donde enviar el paquete. De esta forma los paquetes "fluyen" de un extremo a otro de la red y se consigue un enrutamiento a mayor velocidad a la vez que se disminuye el retardo y el jitter.

2.1 Arquitectura de MPLS

2.1.1 FEC

Es un conjunto de paquetes con similares características que son reenviados con la misma prioridad a través de un mismo LSP (Label Switched Path). Este grupo de paquetes están todos identificados por la misma etiqueta. Las FECs son una manera de

distinguir un tipo de tráfico de otro. Cada paquete en una red MPLS es asignado a un FEC por única vez en el router de ingreso.

2.1.2 LSP

El LSP es el camino compuesto por uno o varios LSR (Label Switched Router) a través del cual se transmiten todos los paquetes pertenecientes a un determinado FEC. Estos caminos son unidireccionales, es decir, solo transmiten tráfico en un sentido.

MPLS soporta dos opciones para la creación de un LSP: (1) LSP salto a salto, y (2) LSP explícito. Para el establecimiento de un LSP salto a salto, cada nodo elige de forma independiente el siguiente salto para encaminar un FEC determinado y esto generalmente se realiza utilizando el protocolo LDP.

En el caso de un LSP explicito, los LSRs no eligen de forma independiente, sino que un sólo LSR es el que define todos o la mayoría de los LSRs que conforman el LSP. EL LSP explícito puede ser configurado o puede ser determinado dinámicamente por algunos medios, por ejemplo, mediante encaminamiento basado en restricciones. [1]. Un LSP también puede ser llamado como un túnel LSP, porque el tráfico a través de él es opaco a los nodos intermedios a lo largo del camino [8].

2.1.3 LSR

Los LSR son todos aquellos routers que se encuentran dentro de una red MPLS. A diferencia de un router convencional, estos routers reenvían los paquetes en función de las etiquetas de los paquetes recibidos, y no en función de la dirección IP de destino. En una red MPLS podemos encontrar dos tipos de LSR:

LER (*Label Edge Router*): Los LER son los routers frontera que operan en los bordes de una red MPLS. Estos routers son los encargados de convertir los paquetes IP en paquetes MPLS, o viceversa. Dependiendo de esta función, podemos diferenciar entre los router de ingreso (upstream) y los router de salida (downstream). Los primeros se sitúan en la entrada de la red y se encargan de asignar un FEC a los paquetes que reciben y de etiquetarlos para que lleguen a su destino. Los routers de salida son los encargados de hacer la acción contraria, eliminar la etiqueta, y se sitúan al final de la red.

Core Router: Estos son los routers que forman el núcleo de la red y permiten el tránsito de los paquetes hacia su destino.

2.1.4 Cabecera MPLS

La cabecera MPLS se ubica entre las cabeceras de nivel 2 y 3 [48] y contiene los siguientes campos como se observa en la Figura 1.

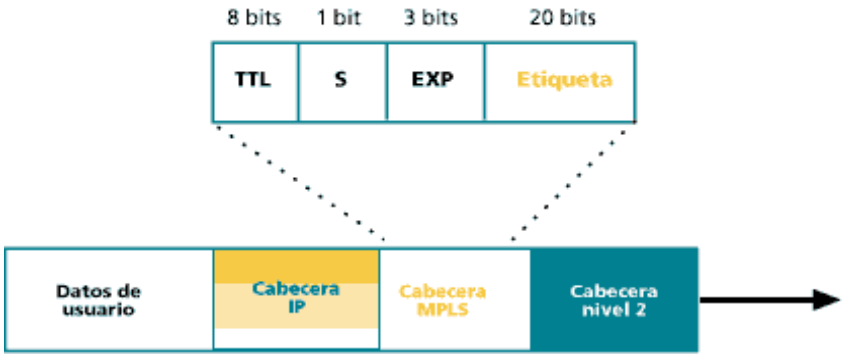

Figura 1. Cabecera MPLS

Etiqueta (Label): Identificador local que contiene la información para enrutar un paquete hacia su destino. Las etiquetas se utilizan en los routers para diferenciar entre los distintos FECs (Forward Equivalance Class).

EXP: Indica la calidad de servicio que requiere el paquete. Anteriormente era denominado CoS (Class of Service) pero ahora se considera un campo experimental.

S: Indica si existe más de una etiqueta, de modo que el nodo MPLS tratará siempre la que esté más alto en la pila.

TTL: Tiene el mismo significado que en IP.

2.1.5 Protocolos de distribución de etiquetas

Para mapear etiquetas en un LSP es necesario un protocolo de distribución de etiquetas. Existen diversas propuestas de protocolos para realizar dicha función como son:

- Protocolo de distribución de etiquetas LDP [3].
- Protocolo de reserva de recursos con extensiones de Ingeniería de tráfico RSVP -TE [8].
- Protocolo de enrutamiento basado en restricciones LDP (CR-LDP) [51].
- Multi-protocolo BGP [11].

LDP es un protocolo de distribución de etiquetas y enrutamiento implícito utilizado para configuración y establecimiento de LSPs salto a salto o también llamados "control-driven LSPs" definido por la IETF. El protocolo LDP funciona sobre TCP y usa la información de enrutamiento subyacente proporcionada por un IGP con el fin de enviar paquetes etiquetados. LDP asocia un FEC con cada camino LSP que se crea, y posteriormente intercambia y distribuye esta información de asociación de las etiquetas entre dos LSR vecinos. Esta asociación es bidireccional y permite que un LSR aprenda del otro.

Por otro lado, entre los protocolos de enrutamiento explícito más comunes encontramos al protocolo LDP de Ruta Restringida (CR-LDP) y al Protocolo de Reservación de Recursos con Ingeniería de Tráfico (RSVP-TE). El primero de estos protocolos es una extensión del protocolo LDP. CR-LDP incorpora características para poder realizar Ingeniería de Tráfico, como la capacidad de poder señalizar caminos explícitos.

RSVP-TE opera de manera similar que CR-LDP, pues permite negociar un LSP punto a punto que garantice un nivel de servicio de extremo a extremo [2]. El protocolo es una extensión de la versión original RSVP [50]. Más adelante en este documento se realizara una descripción más completa de RSVP-TE.

2.1.6 Funcionamiento de una red MPLS

El funcionamiento del protocolo MPLS debe seguir los siguientes pasos:

1. Creación y distribución de etiquetas.
2. Creación de tablas en cada router.
3. Creación de LSPs.
4. Agregar etiquetas a los paquetes con la información de la tabla.
5. Envío del paquete.

Las operaciones de MPLS se ejecutan en los LSR. Los routers LER, también llamados router de ingreso como se explicó anteriormente, operan como la principal interface entre la red MPLS y la tecnología de Capa 2 existente. Los routers LER son los encargados de colocar la etiqueta MPLS cuando los paquetes ingresan a la red MPLS provenientes de redes externas [5] y se utilizan protocolos de distribución de etiquetas para compartir la información de las etiquetas entre los distintos LSRs.

Cada LSR construye una tabla de etiquetas LIB (*Label Information Base*) a medida que recibe la información de las etiquetas. La tabla LIB es donde se especifica el mapeo de cada etiqueta con un interfaz, tanto de entrada como de salida. En el ejemplo de la

Figura 2, un paquete de entrada por la interfaz 2 con la etiqueta 51, se redirigiría a la interfaz 5 con la etiqueta 37.

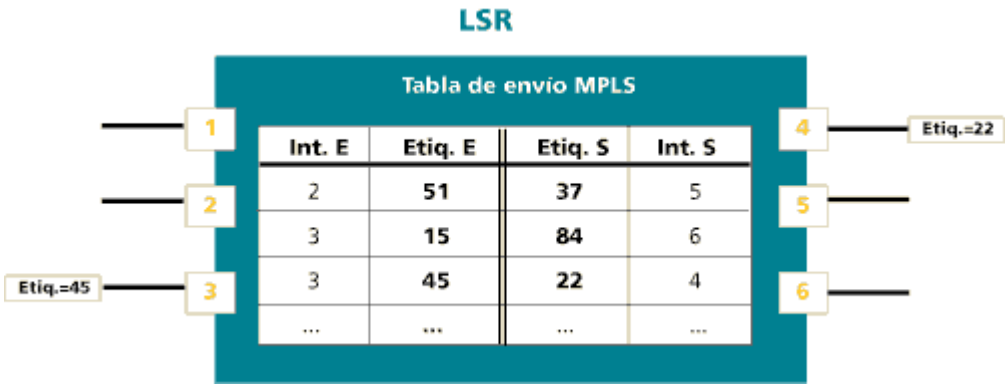

Figura 2. Label Information Base (LIB)

Los LSR construyen también la tabla LFIB (Label Forwarding Information Base) mediante los protocolos de distribución de etiquetas utilizados en el plano de datos y se utiliza para el reenvío de paquetes dentro de un dominio MPLS [2].

El siguiente paso es la creación de los LSP, los cuales se crean en orden inverso a la trayectoria del paquete, lo que significa que el LSP se crea en el Nodo Destino hacia el Nodo Origen.

Para la transmisión de un paquete sobre la red MPLS, una vez que el paquete ya está etiquetado, se envía al siguiente salto LSR usando la tabla LFIB. Este paquete va saltando de LSR en LSR basándose en la tabla LFIB de cada router. Para realizar el reenvío de paquetes en una red MPLS los LSR utilizan las siguientes operaciones:

- **SWAP**: La etiqueta se intercambia por una nueva etiqueta y el camino asociado con la nueva etiqueta es utilizado para el reenvío de paquetes.
- **PUSH**: Una nueva etiqueta es colocado sobre la etiqueta existente, lo que se denomina pila de etiquetas. Esto ayuda a encapsular el paquete en otra capa de MPLS. Esta operación se utiliza en MPLS VPN.
- **POP**: La etiqueta se elimina del paquete. En este punto, el paquete ya puede ser salir de la red MPLS y rutearse como cualquier otro paquete IP.

Cuando un LSR recibe un paquete etiquetado, a la etiqueta superior se le debe aplicar una de las operaciones indicadas, de acuerdo a lo que determine el LSR basándose en la tabla LFIB.

Normalmente lo que hacen estos routers es hacer un *SWAP* de la etiqueta. Finalmente, el paquete llega el router LER de salida, el cual es el encargado de quitar la última etiqueta realizando una operación POP y enviar el paquete hacia su destino por enrutamiento IP convencional. Puede acordarse también entre el último y el penúltimo nodo que sea el penúltimo quien retire la etiqueta, con lo cual puede evitarse en el último nodo dos búsquedas en las tablas de envío, primero en la tabla de MPLS y luego en IP. La Figura 3 muestra el esquema del funcionamiento de una red MPLS.

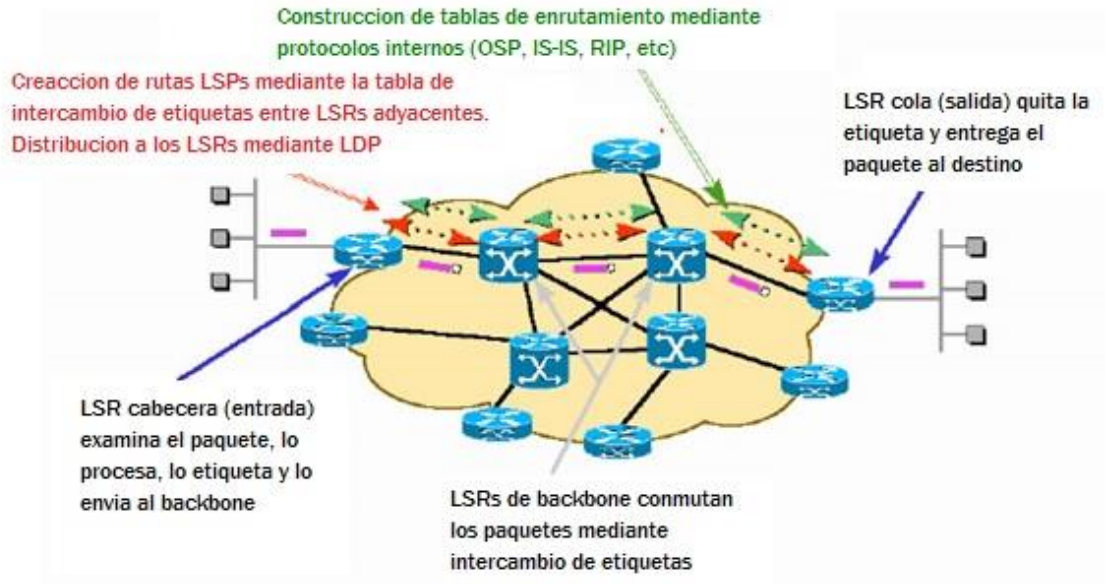

Figura 3. Operación de MPLS

2.2 Aplicaciones de MPLS

2.2.1 Ingeniería de Tráfico

En el esquema de enrutamiento tradicional los paquetes que se transmiten entre dos extremos son enviados utilizando la ruta más corta, se utiliza esta técnica aun cuando existen rutas alternas disponibles que podrían ayudar a solucionar problemas de congestión en la red.

MPLS utiliza estas rutas alternas de manera más eficiente, permitiendo que el tráfico pueda ser enrutado utilizando múltiples caminos y garantizando la disponibilidad de ancho de banda a los flujos de datos. En MPLS los LSP pueden ser optimizados y predefinidos mediante MPLS-TE (MPLS Traffic Engineering). El desarrollo de TE proviene de la necesidad de identificar las rutas requeridas de una manera dinámica debido a la naturaleza aleatoria de Internet [22].

2.2.2 Gestión de caminos

La Gestión de caminos lleva a cabo todas las tareas relacionadas con el mantenimiento y la creación de túneles LSP. Determina las normas relativas a la selección de nuevas rutas y da soporte para los LSPs ya establecidos. El establecimiento de rutas y la selección de rutas son los elementos básicos de la Gestión de caminos; establecimiento de rutas define los LSPs a través de protocolos de señalización que también funcionan como protocolos de distribución de etiquetas mientras que la selección de rutas habilita los túneles LSPs en el nodo origen del túnel.

2.2.3 Redes privadas virtuales (VPN)

La RFC 2547 define un mecanismo el cual permite a los proveedores de servicios utilizar su backbone IP/MPLS para proporcionar servicios de VPN a sus clientes. Los dispositivos que conforman la arquitectura MPLS VPN se los identifica de acuerdo a su ubicación dentro de la red:

- El router del cliente que conecta la red del cliente con la red del proveedor de servicios se denomina "Costumer Edge Router" (CE-router) o también llamado CPE.
- Los dispositivos del proveedor de servicios que se conectan a los routers del cliente se denominan "Provider Edge" (PE).
- Y finalmente aquellos dispositivos que únicamente proveen transporte de datos a través de la red troncal del proveedor de servicios y no tienen clientes directamente conectados, se denominan "Provider devices" (P).

El dispositivo de borde cliente (CE) es un router que establece adyacencia con el PE directamente conectado. Después de establecer adyacencia el router CE anuncia las rutas locales del sitio VPN y aprende rutas remotas desde el PE mediante algún protocolo de enrutamiento o mediante rutas estáticas. Los PE mantienen tablas de encaminamiento distintas (VRF) para cada cliente, lo cual permite que la información de enrutamiento de un cliente y otro se mantenga separada.

Los routers de core P funcionan como LSR de MPLS enviando y conmutando etiquetas, no tienen conocimiento de la información de enrutamiento de las VRF. Para el intercambio de información de enrutamiento de las VRFs, los routers PE utilizan el protocolo MP-BGP [44].

MPLS VPN a través de un marco orientado a conexión permite a los proveedores proveer servicios de VPN sobre una infraestructura IP normalmente no orientado a

conexión. MPLS VPN es un elegante sustituto de los circuitos virtuales permanentes de Frame Relay. La principal ventaja del modelo de MPLS VPN sobre Frame Relay es que MPLS VPN es altamente escalable.

2.2.4 Soporte Multiprotocolo

MPLS tiene características muy atractivas como el soporte de múltiples protocolos. Debido a su naturaleza multiprotocolo, puede soportar calidad de servicio en las redes actuales y además provee soluciones para realizar el mejor uso de los recursos existentes en la red.

3 Ingeniería de Tráfico

La Ingeniería de Tráfico (TE) es una disciplina que procura la optimización del rendimiento de las redes operativas mediante la aplicación de tecnologías y los principios científicos en la medición, caracterización, modelado, y control del tráfico que circula por la red. Las mejoras del rendimiento de una red operacional mediante un eficiente manejo del tráfico y modo de utilización de recursos, son los principales objetivos de TE. Una ventaja práctica de la aplicación sistemática de los conceptos de Ingeniería de Tráfico a las redes operacionales es que ayuda a identificar y estructurar las metas y prioridades en términos de mejora de la calidad de servicio dado a los usuarios finales de los servicios de la red.

La ingeniería de tráfico se subdivide en dos ramas principalmente diferenciadas por sus objetivos:

Orientada a tráfico: ésta rama tiene como prioridad la mejora de los indicadores relativos al transporte de datos, como por ejemplo: minimizar la pérdida de paquetes, minimizar el retardo, maximizar el rendimiento, obtener distintos niveles de acuerdo para brindar calidad de servicio, etc.

Orientada a recursos: ésta rama se plantea como objetivo, la optimización de la utilización de los recursos de la red, de manera que, no se saturen partes de la red mientras otras permanecen subutilizadas, tomando principalmente el ancho de banda como recurso a optimizar.

Ambas ramas convergen en un objetivo global, que es minimizar la congestión. Un reto fundamental en la operación de una red, especialmente en redes IP públicas a gran escala, es incrementar la eficiencia de la utilización de recursos mientras se minimiza la posibilidad de congestión. En casos de congestión de algunos enlaces, el problema se resolvía a base de añadir más capacidad a los enlaces. Pero mediante el uso de la Ingeniería de Tráfico el objetivo es adaptar los flujos de tráfico a los recursos físicos de la red, equilibrando de forma óptima la utilización de esos recursos, de manera que no haya algunos que estén sobre-utilizados, creando cuellos de botella, mientras otros puedan estar subutilizados [12].

En resumen, TE provee de capacidades para realizar lo siguiente:

- Mapear caminos primarios alrededor de conocidos cuellos de botella o puntos de congestionamiento en la red.

- Lograr un uso más eficiente del ancho de banda disponible, asegurando que ciertos recursos de la red no se vuelvan sobre-utilizados, mientras otros recursos son sub-utilizados a lo largo de potenciales caminos alternativos.
- Maximizar la eficiencia operacional.
- Mejorar las características de la performance del tráfico orientado de la red, minimizando la pérdida de paquetes, minimizando períodos prolongados de congestión y maximizando el rendimiento.
- Mejorar las características estadísticas de los límites de la performance de la red (como ser tasa de pérdidas, variación del retardo y retardo de transferencia).
- Proveer de un control preciso sobre cómo el tráfico es re-enrutado cuando el camino primario se enfrenta con una sola o múltiples fallas.

3.1 Proceso de la ingeniería de tráfico

Antes de profundizar en las aplicaciones de MPLS a la ingeniería de tráfico de redes IP, es importante revisar el modelo del proceso de la ingeniería de tráfico. El modelo del proceso representa las diferentes fases del ciclo de vida de la ingeniería de tráfico en un contexto operacional. Hay cuatro principales fases de este modelo de proceso que se pueden apreciar en la Figura 4 y son: fase de formulación de políticas, fase de recopilación de datos, fase de análisis y caracterización y fase de optimización del rendimiento.

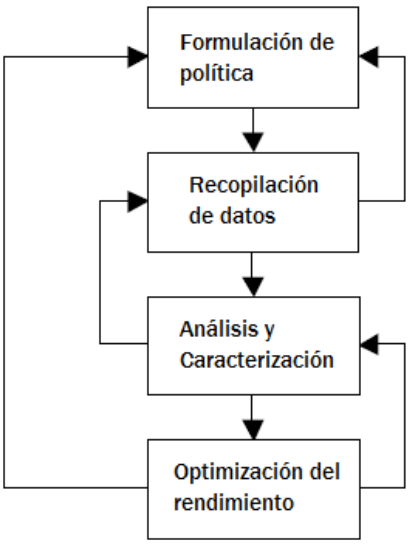

Figura 4. Modelo del proceso de la Ingeniería de Trafico

3.1.1 Fase de Formulación de políticas

La Ingeniería de tráfico eficaz requiere la formulación de una política de control apropiada. Generalmente las políticas dependerán del contexto de red, el modelo de negocio, la estructura de costo, políticas imperantes, y los criterios de optimización.

En el contexto de la ingeniería de tráfico basada en MPLS, habrá de definirse si se realiza ingeniería de tráfico estratégica o táctica. La Ingeniería de tráfico estratégica implica una planificación cuidadosa de la topología virtual de los LSPs, además de considerar cuidadosamente los patrones de tráfico previstos en el futuro para llegar a un plan evolutivo que tenga en cuenta las demandas de tráfico existentes y futuras. La Ingeniería de tráfico táctica se enfoca en la optimización del rendimiento de red mediante el establecimiento y la gestión explícita de LSPs si se requiere hacer frente a problemas de rendimiento específicos [7].

3.1.2 Fase de recopilación de datos

Durante la fase de adquisición de datos, se recogen estadísticas empíricas de la red operativa a través de un sistema de medición. Estas estadísticas capturan características operativas, tales como patrones de tráfico, utilización del enlace, las tendencias de tráfico y estadísticas de paquetes perdidos. Cuando los datos empíricos no se pueden obtener, se pueden utilizar modelos matemáticos. La fase de adquisición de datos es esencialmente la componente de retroalimentación en el modelo del proceso de ingeniería de tráfico.

En el contexto MPLS, la adquisición de datos puede implicar la obtención de estadísticas de desempeño y de fallo asociado con los LSPs, la medición de las rutas recorridas por flujos de tráfico específicos, y la medición de las estadísticas de tráfico entre nodos específicos de la red [7].

3.1.3 Fase de análisis y caracterización

Implica análisis y caracterización del tráfico derivada de la fase de medición. Es esencialmente el aspecto de evaluación del rendimiento de la Ingeniería de tráfico. Uno de los objetivos de la fase de análisis y caracterización es comprender la causa raíz del comportamiento anómalo de la red.

La Ingeniería de tráfico en redes de gran tamaño requiere de un complejo esfuerzo. Por lo tanto, hay una necesidad de realizar análisis fuera de línea (offline) y el uso de herramientas de simulación para apoyar la función de la Ingeniería de tráfico. Las herramientas pueden incluir varios modelos matemáticos y técnicas de optimización,

modelos de recursos, modelos de tráfico, modelos de colas, modelos de series de tiempo, modelos de análisis de enrutamiento, modelos de dimensionamiento de recursos y muchos otros [7].

3.1.4 Fase de optimización del rendimiento

Esta fase implica la aplicación de un apropiado proceso de toma de decisiones para seleccionar el mejor curso de acción para mejorar el rendimiento de la red. La optimización desde el punto de vista de la Ingeniería de tráfico implica un proceso continuo e iterativo de mejora del rendimiento de la red.

En el contexto de MPLS, la fase de optimización puede implicar: (1) la creación de nuevos LSPs y control cuidadoso de las rutas utilizando un apropiado mecanismo de selección de caminos; (2) re-enrutamiento de los LSPs para lograr una distribución del tráfico más equilibrado; (3) desactivación y reactivación de un LSP existente; (4) modificar los parámetros de los LSPs para controlar sus características de comportamiento; (5) la modificación de los atributos asociados a los recursos de red para influir en la colocación de LSPs, entre otros. Esta fase puede implicar también actividades como la modificación de los parámetros de enrutamiento y protocolos de señalización [7].

3.2 Clasificación de los sistemas de Ingeniería de Tráfico

En [9] se presenta una taxonomía de los sistemas de TE, construida sobre la base de estilos y puntos de vista de TE, las cuales se presentan a continuación:

- Dependiente del tiempo vs Dependiente de Estado vs Dependiente de eventos.
- Fuera de línea (offline) vs En línea (online)
- Centralizada vs Distribuida
- Metodologías de TE basado en información local vs Metodologías de TE basado en información global
- Prescriptiva vs Descriptiva
- Bucle abierto vs Bucle cerrado
- Táctica vs Estratégica

La Ingeniería de tráfico requiere el cálculo de los planes de enrutamiento. Este cálculo puede ser realizado *offline* u *online.*

Un sistema de planeamiento y análisis offline es adecuado para escenarios en los que los planes de enrutamiento no necesitan ser ejecutados en tiempo real y examinan

en forma simultánea las restricciones de recursos de cada enlace y los requerimientos de cada LSP. Si bien el acercamiento offline puede tardar varias horas en completarse, realiza cálculos globales comparando los resultados de cada cálculo y selecciona entonces la mejor solución de la red tomada como un conjunto. Una herramienta de proyección y análisis offline es necesaria si se quiere optimizar la TE globalmente [10].

Por otro lado, en los sistemas online se toma en consideración las restricciones de los recursos y se va calculando un LSP a la vez, a medida que van llegando las demandas. Esto implica que el orden en que los LSPs son calculados es muy importante, ya que depende de los LSPs ya establecidos, por dónde se dirigirá cada nuevo LSP que llega. Si se cambiara el orden de llegada de los LSPs, es muy probable que los caminos elegidos para establecerlos también cambien. De esta manera, los LSPs que se calculan primero tienen más recursos disponibles para utilizar que los que llegan más tarde. A diferencia del cómputo *offline*, el cómputo de planes de enrutamiento *online* está dirigido a cálculos sencillos y rápidos en la selección de rutas, mejorar la asignación de recursos y realizar balanceo de carga.

3.3 Limitaciones del enrutamiento IP tradicional

En el enrutamiento IP convencional, cada router toma decisiones de enrutamiento independientes basándose únicamente en la dirección IP destino que se encuentra en el encabezado de los paquetes IP. El principal problema con este tipo de enrutamiento es que no considera requerimientos de capacidad y tráfico que requieren los flujos de datos. El resultado es que algunos segmentos de la red pueden llegar a congestionarse mientras existen rutas alternativas que son infrautilizadas. Incluso en situaciones de congestionamiento de red, los protocolos de enrutamiento tradicionales continúan reenviando tráfico por el camino original o "ruta más corta" hasta que se produce pérdida de paquetes, retardos y jitter que afectan especialmente a las aplicaciones sensibles a retardo como la voz sobre IP [43].

Es difícil realizar Ingeniería de tráfico en redes IP [43]. Para poder enrutar flujos de datos de aplicaciones interactivas que requieren bajo retardo y perdidas, es clara la necesidad de utilizar los recursos de la red de forma más eficiente y el proceso mediante el cual se logra este objetivo se denomina Ingeniería de tráfico.

3.4 Ventajas de MPLS para la Ingeniería de Tráfico

Un enfoque popular para eludir las insuficiencias de enrutamiento de los IGP es mediante el uso de un modelo *overlay*, tales como IP sobre ATM. El modelo *overlay* [31] amplía las opciones de diseño, permitiendo implementar una topología de red virtual sobre una topología de red física. Dicha topología virtual se construye a partir de circuitos virtuales que son considerados como enlaces físicos para el IGP.

MPLS es estratégicamente importante para la ingeniería de tráfico, ya que puede proporcionar la mayoría de la funcionalidades disponibles en el modelo overlay de una manera integrada y a un costo mucho más bajo que las actuales alternativas en competencia. Algunas de las ventajas que ofrece MPLS en comparación con el modelo *overlay* incluyen:

- Menos elementos de red.
- Menos costos de operación.
- Mayor confiabilidad ya que existen menos elementos de red en una determinada ruta.
- Potencialmente menos latencia.
- Arquitecturas de red simplificadas.

La RFC 2702 de la IETF [6] describe un conjunto de capacidades que le permiten a MPLS convertirse en un medio efectivo para implementar varias políticas de ingeniería de tráfico en redes IP. En [36] se encuentra una descripción resumida de estas capacidades:

- MPLS permite crear fácilmente LSPs sin el paradigma del ruteo basado en la dirección IP destino, a través de acciones administrativas manuales o mediante la acción automática de los protocolos subyacentes.
- Los LSPs pueden ser mantenidos de manera eficiente.
- Se pueden crear troncales de tráfico (TT) y se las puede mapear a los LSPs. Un TT es una agregación de flujos de tráfico que pertenecen a la misma clase y que se envían a través de un camino común [9].
- Un conjunto de atributos pueden ser asociados a los TTs para modelar su comportamiento.

- Se pueden asociar un conjunto de atributos a los recursos a fin de restringir el establecimiento de LSPs y TTs a través de ellos.
- MPLS permite tanto la agregación de tráfico y desagregación mientras que el clásico enrutamiento basado en la IP destino permite solamente agregación.
- Es relativamente fácil la integración de un marco enrutamiento basado en restricciones con MPLS.
- Una buena implementación de MPLS puede ofrecer un *overhead* significativamente inferior que otras alternativas para la ingeniería de tráfico.

3.5 Requisitos para el soporte de TE en redes MPLS

Las capacidades funcionales requeridas para un completo soporte de TE sobre redes MPLS de gran tamaño, incluyen el soporte de atributos de troncales de tráfico, atributos de recursos y enrutamiento basado en restricciones.

3.5.1 Troncales de tráfico y sus atributos

Una troncal de tráfico (TT) es una agregación de varios flujos de tráfico de la misma clase al que se le pueden asociar características específicas y que se transmite a través de un mismo LSP, al igual que el LSP los TTs son unidireccionales. A continuación se describen algunos atributos de los TTs que en conjunto especifican sus características de comportamiento [36]:

- Atributos de Parámetros de Trafico como tasa máxima, tasa promedio, tamaños ráfaga (burst size), etc.
- Atributos de Selección y Mantenimiento de Ruta. Estos incluyen, por ejemplo, atributos de preferencia de ruta en caso de múltiples caminos definidos, atributo de adaptabilidad que define si una ruta podría ser re-calculada en caso de cambios en el estado de la red, y atributos de distribución de carga.
- Atributo de Prioridad que define la importancia relativa de un TT.
- Atributo de Preferencia (Preempt Atribute), atributo que determina si un TT puede desplazar a otro TT de un camino dado.
- Atributo de Adaptabilidad (Resilience Atribute) que determina lo que ocurre bajo una condición de fallo, esto es, si un TT no es re-enrutado, o si lo es, en el caso de que existan suficientes recursos en alguna otra ruta, etc.

- Atributos de Vigilancia que determinan la acción a tomar cuando un TT no cumple ciertos requisitos y se convierte en no conforme. Las posibles acciones podrían ser descartar los paquetes o tratarlos como tráfico de máximo esfuerzo (best effort).

3.5.2 Atributos de Recursos

Se trata de un conjunto de atributos asociados a los recursos que condicionan o restringen el uso de dichos recursos a determinados TTs [36].

- Multiplicador de asignación máxima (MAM) de un recurso, es un atributo configurable administrativamente que determina la proporción del recurso que está disponible para su asignación a los TT. Este atributo es mayormente aplicado al ancho de banda del enlace.

- Atributo Clase de Recursos se utiliza para agrupar los recursos en conjuntos. Puede ser utilizado para implementar muchas de las políticas de optimización del rendimiento en materia de tráfico y recursos. Por ejemplo, si se consideran los enlaces como recursos, incluso enlaces no vecinos pueden ser añadidos en una clase y varias políticas podrían definirse como políticas específicas de inclusión o exclusión para un tráfico específico, o especificar preferencias para la colocación de los TT entre varios conjuntos de recursos.

3.6 Enrutamiento basado en restricciones

El enrutamiento basado en restricciones (CBR) es una de las capacidades funcionales más importantes para la operación de MPLS-TE. El enrutamiento basado en restricciones selecciona la mejor ruta que obedece a las restricciones establecidas de manera que sea óptimo respecto a alguna métrica escalar (por ejemplo el minimizar la cantidad de saltos o una métrica administrativa) [21]. Dichas restricciones son impuestas, por un lado, por políticas de enrutamiento que administran, gestionan y controlan el acceso a los recursos de la red, y por otro lado, por requisitos de calidad de servicio dados por el uso del ancho de banda, retardos, jitter y pérdidas de paquetes [38].

Para lograr estos objetivos, se utiliza el algoritmo CSPF (*Constrained Shortest Path First*) [32], que es una extensión del algoritmo SPF (*Shortest Path First*). El algoritmo CSPF requiere que el LSR que realiza la computación del camino tenga información sobre todos los enlaces en la red. Esto impone una restricción en el tipo de protocolo de enrutamiento que se puede usar, es decir, se deben usar protocolos de estado de enlace como IS-IS u OSPF. CSPF integra información de topología de estado de enlace que

aprende de los IGP y la mantiene en la base de datos de ingeniería de tráfico (TED). La información almacenada en la TED incluye atributos asociados con el estado de los recursos de la red (tales como ancho de banda completo de los enlaces, ancho de banda reservado, ancho de banda de disponible, y color de enlace). En la Figura 5 se puede apreciar el proceso total de computación del algoritmo CSPF.

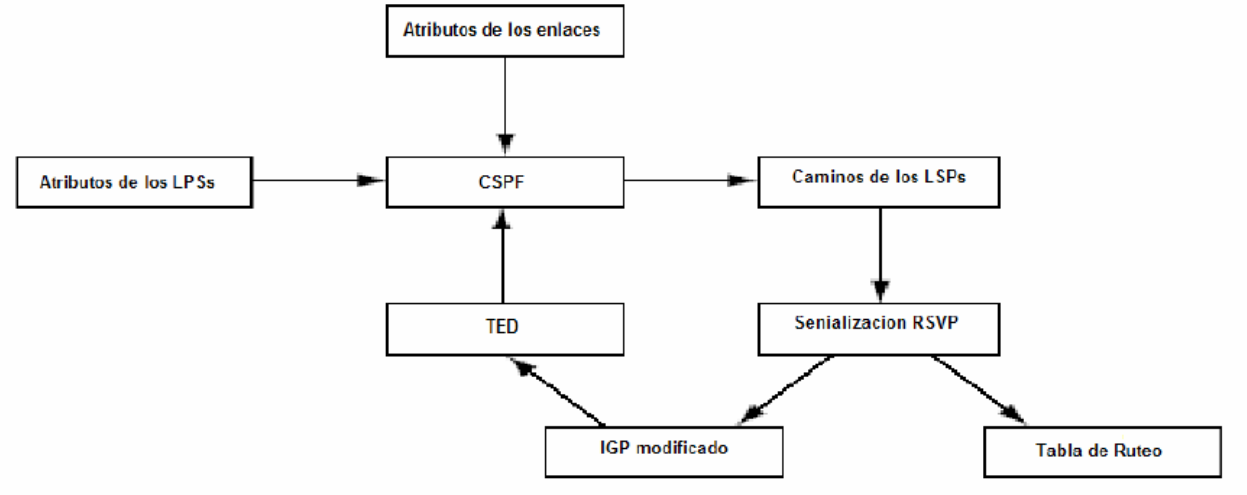

Figura 5. Proceso de computación del CSPF

Uno de los mecanismos restantes necesarios para soportar CBR, es la capacidad de ruteo explícito que es provista por MPLS [15]. Una ruta explícita puede ser una lista de direcciones IP, puede usarse también en una ruta explícita el concepto de "nodo abstracto", que es una colección de nodos que se presentan como si fueran uno solo. Un ejemplo de nodo abstracto puede ser un sistema autónomo (AS).

En MPLS-TE, el LSR de ingreso calcula una ruta que satisfaga un conjunto de restricciones en el estado actual de la red utilizando un protocolo de enrutamiento basado en restricciones. Para encontrar una la ruta basada en restricciones se debe correr un algoritmo de enrutamiento basado en restricciones. Dos protocolos estuvieron estandarizados para el enrutamiento basado en restricción en MPLS, uno de ellos, CR-LDP, y otro es RSVP-TE, pero de acuerdo a la RFC 3468 [4], la IETF establece centrar sus esfuerzos en RSVP-TE como protocolo para la señalización de Ingeniería de tráfico, por ello no se profundizara en este trabajo sobre CR-LDP.

3.7 RSVP-TE

Después de un LSP se calcula con CSPF, ese camino necesita señalización a través de la red [40]. Esta señalización se realiza utilizando RSVP, junto con extensiones de RSVP para MPLS-TE.

RSVP [50] es un protocolo de la capa de trasporte para señalización y fue desarrollado por el grupo de trabajo del Internet de Servicios Integrados (IntServ) de la IETF. Proporciona garantías de calidad de servicio de extremo a extremo mediante el aprovisionamiento del ancho de banda requerido para la transferencia de datos, permitiendo alcanzar un retardo mínimo. RSVP fue diseñado para interoperar con los actuales y futuros protocolos de enrutamiento para optimizar el uso de recursos y lograr un mejor rendimiento de la red [6].

RSVP-TE es una extensión del protocolo original RSVP que fue diseñado para ejecutar distribución de etiquetas sobre MPLS, RSVP-TE soporta además la creación de rutas explícitas con o sin reserva de recursos. Una de las características adicionales más importante de este protocolo es que permite el re-enrutamiento de los túneles LSP, con el fin de dar una solución ante caídas de red, congestión y cuellos de botella [34]. La Tabla 1 muestra una lista de los diferentes tipos de mensajes definidos para RSVP. En la RFC 3209 se encuentra la descripción completa del funcionamiento de RSVP-TE.

Tabla 1. Tipos de mensajes RSVP [40]

Tipo de Mensaje	Descripción
Path	Usado para establecer y mantener reservaciones.
Resv (short for Reservation)	Se envía en respuesta a un mensaje PATH para establecer y mantener reservaciones.
PathTear	Es análogo a PATH, pero se usa para remover reservaciones.
ResvTear	Es análogo a RESV, pero se usa para remover reservaciones.
PathErr	Lo envía el receptor de un mensaje PATH en caso de detector un error en el mensaje PATH.
ResvErr	Lo envía el receptor de un mensaje RESV en caso de detector un error en el mensaje PATH.
ResvConf	Se envía opcionalmente de vuelta al que envía un mensaje RESV para confirmar que la reservación se ha completado con éxito.
Hello	Una extensión definida en RFC 3209 que permite el enviar de paquetes keepalives entre dos vecinos RSVP directamente conectados.

3.7.1 ¿Cómo crea RSVP-TE un LSP explícito?

En la Figura 6 se observa un ejemplo del proceso de reserva del protocolo RSVP entre los routers "RtrA" y "RtrF". Los pasos se enumeran a continuación:

1. El LER de entrada RtrA, quiere establecer un nuevo LSP hacia el LER destino RtrF. Los parámetros de tráfico determinan por dónde debe pasar la ruta, así que RtrA envía un mensaje PATH con la ruta explícita hacia RtrF y con los parámetros de tráfico que requiere la sesión.

2. Cada nodo LSR de la ruta que recibe el mensaje determina si es el nodo destino para ese LSP, si no lo es, sigue enviando el mensaje PATH hasta que llega a RtrF.

3. Una vez que llega a RtrF, éste determina qué recursos ha de reservar y devuelve un mensaje RESV que distribuirá la etiqueta que ha elegido para ese LSP y contendrá los detalles de la reserva.

4. Los LSRs intermedios emparejan los mensajes PATH y RESV que han recibido según el identificador de LSP, reservan los recursos que indica RESV, asignan una etiqueta para el LSP, rellenan la tabla de envío y envían la nueva etiqueta en otro mensaje RESV.

5. El LER de entrada RtrA, cuando lo recibe, enviará un mensaje de confirmación RESVConf para indicar que se ha establecido el LSP.

6. Después de haberse establecido el LSP se enviarán mensajes periódicos cada 30 segundos para mantener el camino y las reservas.

7. Es importante indicar que si alguno de los LSR que conforman el camino no tuviera los recursos suficientes solicitados en el mensaje PATH inicial, el LSP (túnel TE) no podrá ser establecido.

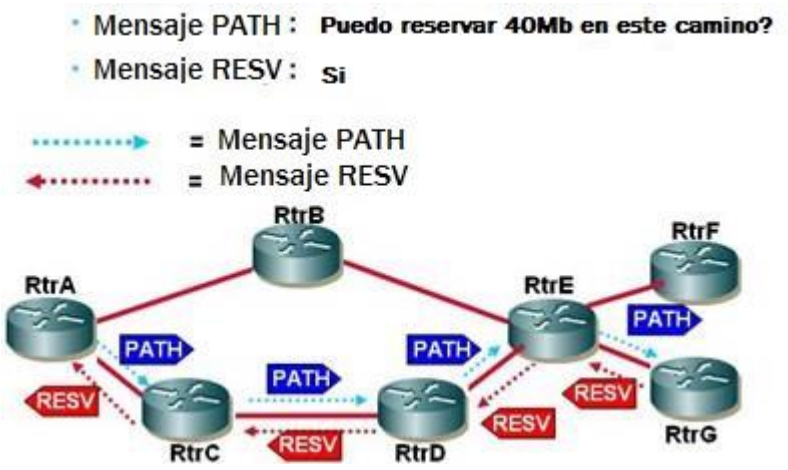

Figura 6. Señalización del LSP utilizando el protocolo RSVP

3.8 Componentes de la Ingeniería de Tráfico en MPLS

Hay cuatro componentes que se pueden destacar dentro de la Ingeniería de Tráfico: la componente del reenvío de paquetes, la componente de distribución de información, la componente de selección de camino y la componente de señalización [9].

3.8.1 Componente de reenvío de paquetes

La componente de reenvío de paquetes es responsable de dirigir un flujo de paquetes IP a lo largo de un camino predeterminado a través de la red.

3.8.2 Componente de distribución de información

Consiste en obtener un conocimiento detallado de la topología de la red así como también información dinámica de la carga en la red. La componente de distribución de información es implementada definiendo extensiones relativamente simples a los IGPs, como son OSPF-TE [28] y IS-IS TE [30], tal que los atributos de los enlaces son incluidos como parte de cada aviso del estado de enlace en cada router. OSPF lleva la información de los atributos en los LSAs Opacos (Type 10 LSA, IS –IS utiliza TLV 22 y TLV 135 para llevar información de ingeniería de tráfico.

Cada router mantiene atributos de los enlaces de la red e información de la topología de la red en la TED. La TED es usada exclusivamente para el cálculo de rutas explícitas, para la ubicación de LSPs a lo largo de la topología física, de manera que el cálculo subsiguiente de la ingeniería de tráfico sea independiente del IGP.

3.8.3 Componente de selección de caminos

Luego que los atributos de los enlaces y la información de la topología han sido inundados por IGP y localizados en la TED, cada router de ingreso utiliza la TED para calcular los caminos de su propio conjunto de LSPs a lo largo del dominio de ruteo. El camino para cada LSP puede ser representado por una ruta explícita. El router de ingreso determina el camino físico para cada LSP aplicando un algoritmo de camino más corto basado en restricciones (*CSPF*) a la información en la TED.

3.8.4 Componente de señalización.

Por último, la componente de señalización es la responsable de que el LSP sea establecido para que sea funcional mediante el intercambio de etiquetas entre los nodos de la red. Esta señalización se realiza utilizando RSVP-TE.

4 Estado del arte de la Ingeniería de Trafico en redes MPLS

4.1 Enfoques analíticos a la ingeniería de tráfico MPLS

De acuerdo a [7] y [19], los principales problemas a los que se enfrenta la ingeniería de Trafico, visto desde una perspectiva operacional son: (1) El enrutamiento basado en restricciones, (2) asignación y particionamiento de tráfico a través de los LSP. Que básicamente consiste en determinar si una petición de conexión o demanda puede ser admitida o no y si es así, cuál sería la ruta más óptima de la conexión a través de la red, (3) Restablecimiento de caminos. Este problema surge debido al fallo y recuperación de uno o más elementos de la red, balanceo de carga o preferencia de ciertas conexiones sobre otras y (4) el problema de diseño de la red y la planificación de capacidad que se ocupa de la determinación de una topología de red óptima para un determinado conjunto de demandas. Estos problemas son inherentes a las fases dos y tres del proceso de la Ingeniería de tráfico.

Esta sección se centra principalmente en algunos estudios de Ingeniería de tráfico en redes MPLS que se han desarrollado para poder resolver los problemas mencionados con anterioridad y dar además una visión actual de la Ingeniería de tráfico.

4.1.1 Algoritmos de enrutamiento basado en restricciones

Los algoritmos de CBR básicos CSPF, *Widest Shortest Path* (WSP) y *Shortest Widest Path* (SWP) [42] son las soluciones de enrutamiento fundamentales que se pueden aplicar a esquemas en línea basados en MPLS-TE.

- **WSP.** Primero selecciona la ruta que tiene un cuello de botella de ancho de banda mayor. El cuello de botella de ancho de banda representa la capacidad mínima no usada de todos los enlaces en el camino. Luego escoge entre los caminos que tienen el menor número de saltos.

- **SWP**. Está optimizado primero por el menor número de saltos y cuando hay múltiples caminos entre ellos, elige el de mayor ancho de banda disponible.

- **MIRA.** Es un algoritmo de enrutamiento de caminos online que intenta minimizar la "interferencia" que provoca el establecimiento de un nuevo camino a potenciales nuevos caminos que son desconocidos, es decir, no hay estimación previa de requerimientos futuros. Se conoce como "interferencia" a la competencia de los LSPs en los enlaces críticos que no cuentan con suficiente ancho de banda disponible para suplir todas las demandas de LSPs [26]. Para comprender su funcionamiento en la Figura 7 consideramos que los nodos ingreso-egreso son (A, G), (B, G), (C, G). Se puede dar la situación en que se necesiten varios LSPs entre

(A,G) y se utiliza el enfoque de "min-hop" los LSPs serán mapeados en el camino con menor cantidad de saltos, saturando los enlaces A-D y D-G y bloqueando futuras demandas entre (B,G) y (C,G), siendo lo ideal un algoritmo que tenga en consideración lo crítico que son estos enlaces en éstas futuras demandas, de modo de mapear la demanda entre (A,G) por A-E-F-G aunque éste camino tenga un mayor número de saltos.

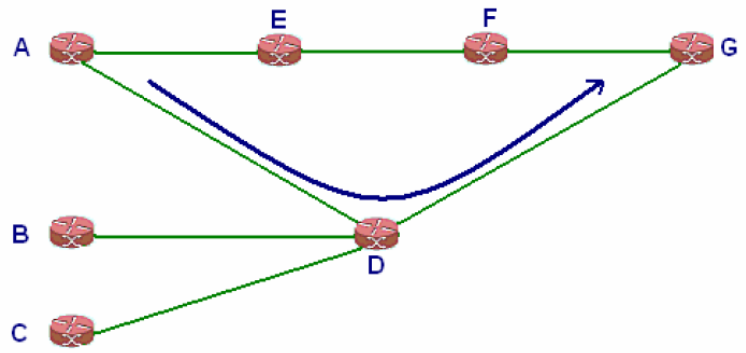

Figura 7. Ejemplo de algoritmo MIRA

Su modelo matemático utiliza el flujo máximo de tráfico que puede ser transmitido entre un par de nodos ingreso-egreso, demostrando mejor rendimiento para enrutamiento de LSPs en comparación con WSP. Este algoritmo trabaja en un nivel de granularidad muy fino, ya que se ejecuta cada vez que un nuevo flujo llega a la red, de aquí que el costo computacional es muy elevado. De acuerdo a Wang et al [48], otro inconveniente de este algoritmo es que para determinar los enlaces críticos toma en cuenta una única pareja de nodos ingreso-egreso, por tanto no es posible estimar puntos críticos de congestión en enlaces que pertenecen a *clusters* de nodos. Esto podría desembocar en el rechazo de peticiones de flujo, aun cuando existen recursos suficientes para cubrir la demanda.

- **WSC.** Wang et al [48] presenta el algoritmo online WSC, que también se basa en MIRA pero adopta un criterio diferente para la identificación de enlaces críticos. El algoritmo propuesto considera no sólo la importancia de los enlaces críticos, sino futuras solicitudes de establecimiento de posibles LSPs. Por otra parte, se incorpora la información de ancho de banda residual del enlace. Los resultados derivados a través de la simulación indican que el algoritmo de Wang et al produce menos rechazos que MIRA, como se puede ver en la Figura 8.

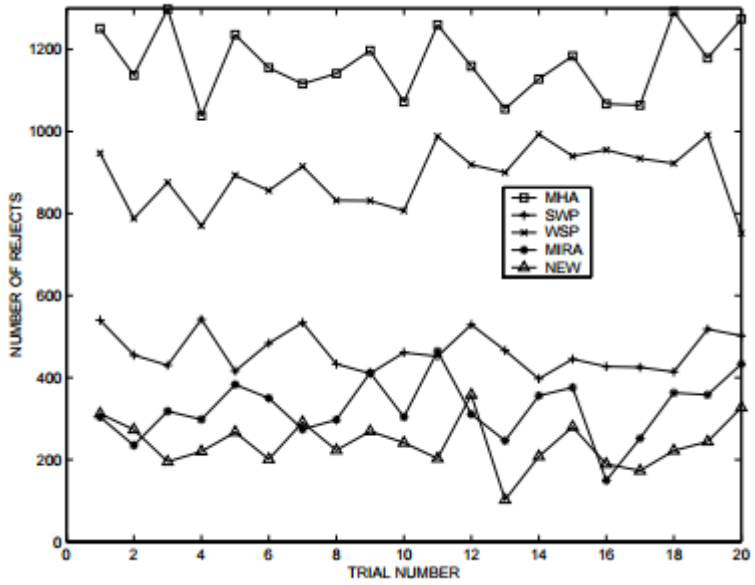

Figura 8. Número de peticiones de configuración de LSP rechazadas en 20 ensayos [48]

- **LMIR** es una propuesta más reciente que las anteriores. Utiliza un algoritmo de Dijkstra modificado para identificar las rutas con menos capacidad y luego estos caminos se utilizan para identificar los enlaces críticos. El número de rutas críticas es un factor clave en este algoritmo. La mejora del tiempo computacional en algunos escenarios de red es un 40 por ciento mejor con respecto al MIRA y WSC [18].

Existen muchas otras propuestas de algoritmos de enrutamiento basado en restricciones aplicados a MPLS-TE que utilizan el concepto de "interferencia" como en [13][27], o algoritmos como RRATE [39] que mediante el uso de un esquema de aprendizaje calcula el orden óptimo de los caminos por los que las solicitudes se pueden dirigir o FCBA [42] que toma decisiones de enrutamiento basado en clases de tráfico y mediante el uso de la lógica difusa. Logra por ejemplo, el mejor valor retardo medio en caso de tráfico intolerante al retardo como la voz en comparación con WSP y SWP.

En esta sección se ha tratado de introducir los principales algoritmos de CBR, ya que muchos de los algoritmos más nuevos se han basado principalmente en los anteriormente mencionados.

4.1.2 Algoritmos de asignación y particionamiento de tráfico

La congestión de red resultante de la asignación ineficiente de recursos puede reducirse mediante la adopción de políticas de balanceo de carga. Cuando la congestión se reduce al mínimo, la pérdida de paquetes disminuye, retardo de tránsito disminuye, y aumenta el rendimiento general de la red. De ésta manera, la percepción de la calidad de servicio que experimentan los usuarios finales se ve significativamente mejorada.

Las redes IP realizan balanceo de carga en enlaces que tienen igual costo, no en enlaces con costo distinto. Cambiar manualmente el costo de un enlace en una red IP para realizar balanceo de carga no es sencillo, podría provocar lazos y afectar el rendimiento en otros sectores de la red. Con MPLS es posible realizar balanceo de carga sobre enlaces que tienen diferente costo, sin el problema de lazos de enrutamiento de las redes IP, debido a que los routers pueden señalizar los LSPs a través de los cuales desean que el tráfico fluya mediante RSVP-TE y dar a conocer esta información a los demás routers que conforman el LSP [40].

El balanceo de carga en MPLS-TE funciona entre múltiples túneles hacia el mismo destino. Si hay más de un túnel a un destino, que comparten entre esos túneles. El administrador de la red puede decidir el porcentaje de ancho de banda que cada LSP puede manejar.

En [45] Ravindra et al, presentan un estudio analítico de algunas de las técnicas de balanceo de carga más conocidas que se encuentran en la literatura, entre ellas, MATE, que a continuación se describe junto con otras técnicas más actuales:

- **MATE**. El objetivo principal de MATE [17] es evitar la congestión de red. Su funcionamiento básico es enviar el trafico entrante de forma adaptativa a través de múltiples LSP pre-construidos de acuerdo a información de retroalimentación sobre la congestión de un camino (retardo, pérdidas). Este paradigma de TE no está relacionado directamente con optimizaciones del enrutamiento, sino más bien con la optimización del tráfico y los recursos a través de la adaptación. MATE es un algoritmo potente pero asume que el tráfico es constante en media y es muy sensible a las variaciones del parámetro. MATE no garantiza requerimientos de QoS al tráfico, simplemente optimiza el costo global de la red [12].

- **OpIATE**. Una contribución clave de OpIATE [46] es explorar el papel de los parámetros de retroalimentación no intrusivos, es decir, los enlaces individuales no están obligados a recoger y enviar información de realimentación como en MATE, por tanto limita la función de distribución de carga de la red a los nodos finales, mientras los nodos del núcleo de la red no participan en el proceso de ingeniería de tráfico.

- **TeXCP** [25] es un enfoque online más reciente para balanceo de tráfico en tiempo real, que busca responder a las demandas de las aplicaciones actuales. Los resultados experimentales demuestran que TeXCP realiza una mejor distribución de carga con tiempos de convergencia menores que MATE.

4.1.3 Restablecimiento de caminos

El tercer problema que trata de resolver MPLS-TE es el restablecimiento de caminos o la restauración rutas de acceso. El funcionamiento fiable de una red es un aspecto importante de la ingeniería de tráfico. Este aspecto es clave para poder cumplir con los requisitos de calidad de servicio que debe ofrecer la red y es un punto clave para los ISPs que tienen que cumplir con acuerdos de nivel de servicio, o también llamados por sus siglas en ingles SLA (Service Level Agreement). Una de las ventajas de MPLS-TE es que permite ofrecer mecanismos de restablecimiento de caminos más eficientes que en las redes IP. Existen dos formas de recuperación de caminos ante un fallo: (1) re-enrutamiento (reroute) y (2) conmutación de protección (protection switching) o también llamado *fast reroute.*

- ***Reroute*** es un modelo que establece una ruta de recuperación después de que ocurre el fallo en la red.

- ***Fast Reroute*** es un modelo donde la ruta de recuperación es pre-calculada y pre-establecida antes de que ocurra cualquier fallo en el camino principal [35], en este caso el resultado en una conmutación más rápida en comparación a *Reroute* [23] y puede ser usado para proteger los LSPs en MPLS-TE ante una falla de nodo o de enlace. *Fast Reroute* permite satisfacer las necesidades de las aplicaciones de tiempo real como la voz sobre IP, con tiempos de conmutación que rodea los 50 milisegundos. Para el soporte de este método de recuperación de caminos fue necesario crear extensiones al protocolo RSVP-TE definidas en RFC 4090 [41].

4.2 Calidad de servicio en MPLS-TE

Durante los últimos años la IETF ha propuesto muchos modelos de servicio y mecanismos para satisfacer la demanda de QoS. El modelo de Servicios Integrados (IntServ) y el modelo de Servicios Diferenciados (DiffServ) son los más notables, pero debido a los problemas de escalabilidad que presenta IntServ [52] ha sido reemplazado por DiffServ.

DiffServ [20] ha sido diseñado para dar diferentes niveles de calidad de servicios a distintos tipos de tráfico. Está basado en un modelo sencillo donde el tráfico que entra en la red se divide en clases mediante marcado. Esta marca llamada DSCP

(Differentiated Service Code Point) usa 6 bits para distinguir una clase de otra. Estos 6 bits se registran en el byte de Type of Service de la cabecera IPv4 y en el byte de Traffic Class en el caso de IPv6. Cada nodo de la red le da al tráfico entrante un tratamiento específico denominado PHB (Per Hop Behavior) [20] dependiendo del tipo de tráfico indicado por la marca DSCP [20].

4.2.1 DS-TE

MPLS-TE no puede proporcionar calidad de servicio por sí mismo, por tanto es necesario complementarlo con otra tecnología capaz de proporcionar dicha función: DiffServ.

DiffServ permite clasificar el tráfico que se envía en un LSP y permite por tanto cumplir con los requisitos de calidad de servicio de un determinado flujo de datos en una red MPLS. Esta estrategia de la Ingeniería de Trafico es lo que se denomina DS-TE [29].

El requisito fundamental para DS-TE es que debe tener la capacidad de soportar diferentes restricciones de ancho de banda para diferentes troncales de tráfico. DS-TE permite asignar una porción de ancho de banda dentro del túnel TE exclusivamente para tráfico que debe ser tratado con calidad de servicio. Esto permite garantizar el límite de este tipo de tráfico dentro del túnel TE.

Para especificar la clase de servicio a la que pertenece cada paquete se utiliza el soporte de MPLS para DiffServ, donde se redefine la cabecera EXP de MPLS para la especificación de dicha clase de servicio [33]. El campo EXP de la cabecera de MPLS es de tres bits, por lo que cada paquete puede pertenecer a una de las 2^3 = 8 clases posibles. Estas clases están definidas en la RFC 3564 como *Class-Type* (CT) y van desde CT0 para el tráfico de menor prioridad o *best effort*, hasta CT7 para el tráfico de mayor prioridad. CT se utiliza para asignación de ancho de banda de enlace, enrutamiento basado restricciones y control de admisión. Por lo tanto, CSPF ha sido modificado para incluir las restricciones de ancho de banda correspondientes a una CT específica.

5 Diseño de escenarios de pruebas

Los experimentos que se proponen en el presente trabajo están enfocados en mostrar la operación de algunas funcionalidades de MPLS-TE que se pueden realizar utilizando mecanismos de ingeniería de tráfico online, a través de ejemplos prácticos en un ambientes intra-dominio. Dichas funcionalidades forman parte de la fase de optimización del rendimiento dentro del proceso de la ingeniería de tráfico.

5.1 Software utilizado

5.1.1 VNX

Para realizar el diseño e implementación de los escenarios de red que permitan mostrar las funcionalidades y beneficios que aporta MPLS a las redes IP, se ha utilizado la herramienta de simulación VNX [16] que es una herramienta de simulación de redes desarrollada por el Departamento de Ingeniería Telemática (DIT) de la Universidad Politécnica de Madrid.

VNX es una herramienta de virtualización de código abierto de uso general diseñada para ayudar a la construcción de bancos de pruebas virtuales de red automáticamente. Permite la definición y la implementación automática de escenarios de red compuestos de máquinas virtuales de diferentes tipos (Linux, Windows, FreeBSD, oliva o Dynamips routers, etc.) interconectados siguiendo una topología de red definida por el usuario. Para el soporte de routers Cisco integra Dynamips y la plataforma de virtualización Olive para routers Juniper.

5.1.2 Iperf

Para realizar mediciones de ancho de banda y de rendimiento en el escenario de pruebas, se utilizó un programa cliente servidor muy sencillo llamado Iperf [24]. Iperf permite medir el ancho de banda mediante pruebas con tráfico TCP y el rendimiento de red utilizando tráfico UDP. Emite reportes de ancho de banda máximo, jitter y pérdida de datagramas.

5.1.3 Wireshark

Es un analizador de protocolos que permite capturar todo el tráfico que pasa a través de una red, por tanto es muy utilizado para análisis y solución de problemas en redes de comunicaciones. Wireshark es un software libre, posee una interfaz gráfica y

muchas opciones de organización y filtrado de información que facilitan el análisis de los datos capturados, a través de los detalles y sumarios por cada paquete [49].

5.2 Hardware utilizado

La herramienta de simulación VNX se ejecuta en una máquina virtual con sistema operativo Ubuntu con 2GB de RAM virtualizada utilizando VMware. El ordenador donde se ejecuta todo este software tiene las siguientes características:

- Marca Sony Vaio
- SO Windows 7
- Intel Core i5 CPU 2.5 GHz
- 6 GB RAM

5.3 Descripción del escenario de pruebas base

El diseño experimental consiste en un entorno de red MPLS emulado utilizando routers Cisco 7206 con versión de ios c7200-advipservicesk9-mz.152-4.S6 y ordenadores con sistema operativo Ubuntu. El diagrama físico de la red utilizada para las pruebas se muestra en la Figura 9 y la configuración de VNX del escenario implementado se incluye en el Anexo 5. El direccionamiento IP utilizado se describe en la Tabla 2.

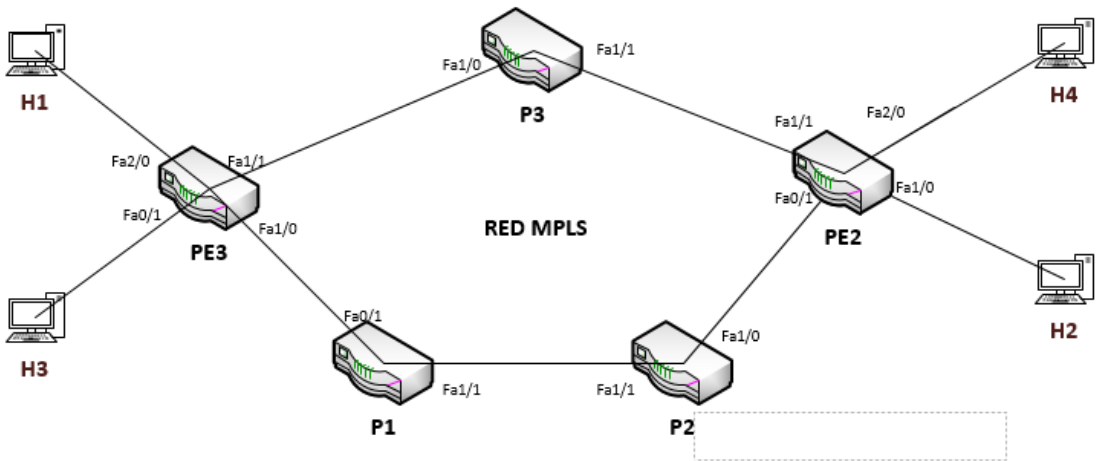

Figura 9. Diagrama de pruebas base

Tabla 2. Direccionamiento IP utilizado

Host	Interface	IP	Conecta con
P1	Loopback0	10.101.1.247	
	FastEthernet1/1	10.1.1.5	P2
	FastEthernet0/1	10.1.1.17	PE3
P2	Loopback0	10.101.2.247	
	FastEthernet1/1	10.1.1.6	P1
	FastEthernet1/0	10.1.1.21	PE2
P3	Loopback0	10.201.1.247	
	FastEthernet1/0	10.1.1.14	PE3
	FastEthernet1/1	10.1.1.2	PE2
PE2	Loopback0	10.201.2.247	
	FastEthernet0/1	10.1.1.22	P2
	FastEthernet1/1	10.1.1.1	P3
	FastEthernet1/0	172.16.2.1	H2
PE3	Loopback0	10.201.3.247	
	FastEthernet1/1	10.1.1.13	P3
	FastEthernet1/0	10.1.1.18	P1
	FastEthernet2/0	172.16.1.1	H1
	FastEthernet0/1	172.16.3.1	H3
H1	eth1	172.16.1.2	PE3
H2	eth1	172.16.2.2	PE2
H3	eth1	172.16.3.2	PE3
H4	eth1	172.16.4.2	PE2

Por tratarse de una red MPLS, fue preciso realizar adaptación del valor de MTU soportado. El valor por default de MTU en redes IP convencionales en de 1500 bytes, pero MPLS agrega una cabecera adicional de 4 bytes por etiqueta. Por lo tanto, se decidió establecer un valor de MTU de 1700 bytes en todos los enlaces que son parte del núcleo de red MPLS. En los routers el parámetro se configuró a nivel de interface utilizando el comando "mpls mtu 1700" y para cambiar el MTU en los switches virtuales fue necesario implementar un script sencillo llamado "cambia-mtu.bin" que se incluye en el Anexo 6.

Como se puede ver en la Figura 9 la red consta de cisco routers, P1, P2 y P3 actúan como LSRs y PE2 y PE3 actúan como LERs, es decir, están en la frontera de la red MPLS y permiten la conexión de los CPEs de los clientes, H1, H2, H3 y H4. En todas las pruebas PE3 actúa como router de ingreso y PE2 como router de egreso. Se utilizó OSPF como protocolo de enrutamiento para establecer la conectividad de red y en los hosts H1, H2, H3y H4 se instaló la herramienta Iperf.

En todos los experimentos para simular congestión se utilizó Iperf enviando un volumen de tráfico de 4Mbps entre H1 y H4. Para las pruebas de medición de rendimiento también se utilizó Iperf con los valores de tráfico UDP que vienen definidos por default en la herramienta, esto es, datagramas de 1470 bytes y buffer de 160 Kbytes. Las lecturas promedio de ancho de banda, jitter y pérdida de datagramas, se tomaron utilizando Iperf, realizando 5 pruebas cada una sobre un período de tiempo de 10 segundos.

Como las pruebas se desarrollaron sobre una red emulada, fue preciso medir el rendimiento máximo de la red. Los routers tienen interfaces FastEthernet que soportan un ancho de banda aproximado de 100 Mbps en entornos reales sin congestión. Para ello se realizaron pruebas de rendimiento entre los hosts H2 (servidor) y H3 (cliente) utilizando Iperf y se obtuvo un valor promedio máximo de ancho de banda de 4Mbps sin pérdida de paquetes, como se puede ver en la Figura 10. Este valor será utilizado como parámetro para poder realizar comparaciones cuantitativas de rendimiento.

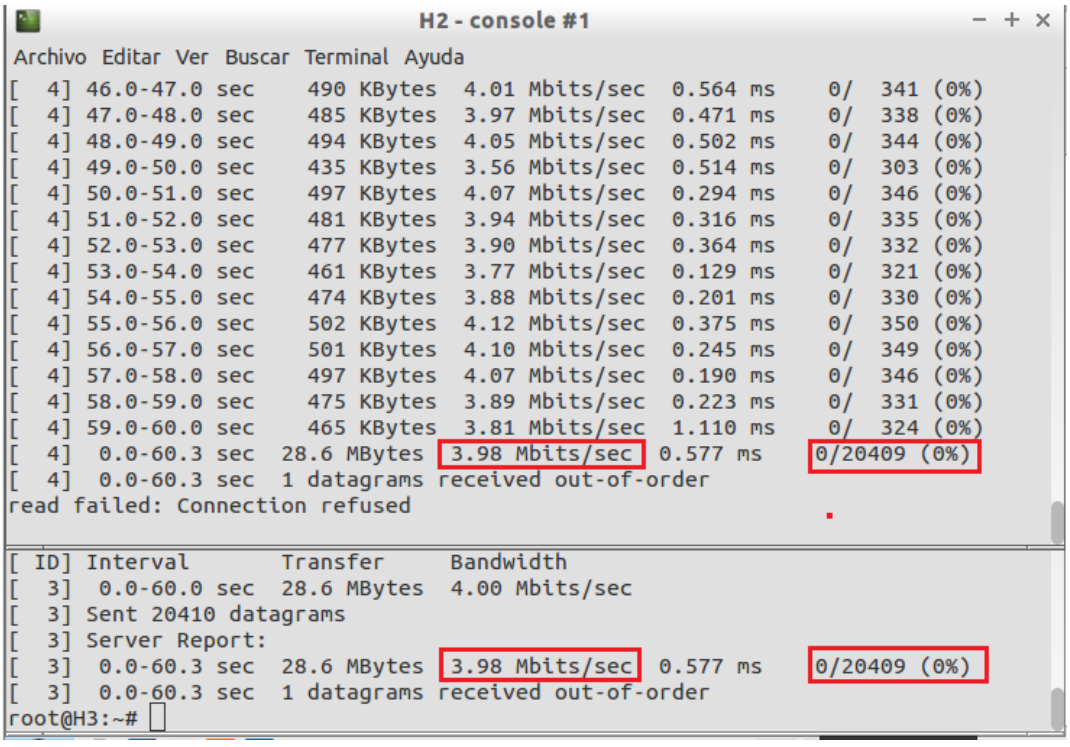

Figura 10. Prueba de rendimiento sobre la red MPLS entre H2 y H3

5.4 Experimentos

5.4.1 Experimento 1: Implementación de Túneles TE para enrutamiento de tráfico

Una de las ventajas principales que nos permite MPLS-TE es poder enrutar tráfico a través de caminos distintos a los calculados por el IGP. En la red mostrada en la Figura 11 se ha simulado congestión mediante Iperf enviando tráfico de 4Mbps entre H1 y H4 sobre la ruta definida por el IGP, que sería el camino PE3-P3-PE2 (ruta más corta).

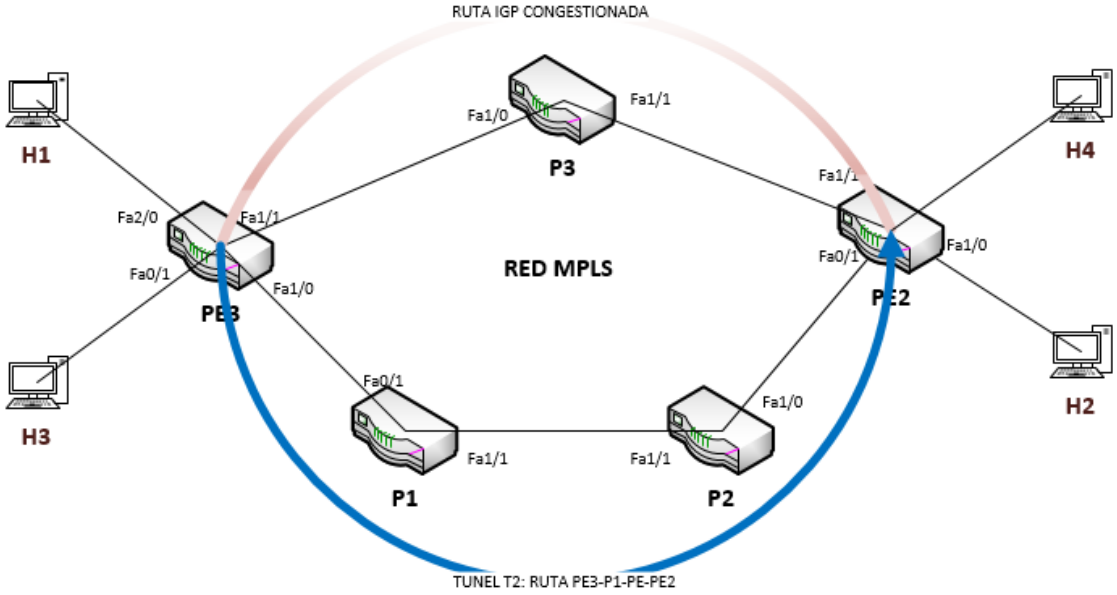

Figura 11. Diagrama Experimento 1

En este experimento se medirá el rendimiento, jitter y porcentaje de pérdida de datagramas en dos escenarios de red: uno sin MPLS-TE y otro con MPLS-TE. Las configuraciones de los routers se encuentran en el Anexo 1.

5.4.1.1 Escenario 1.1: Red MPLS que experimenta congestión sin MPLS-TE.

Como muestra la traza realizada desde H3 hacia H2 y la tabla LFIB de PE3 en la Figura 12, el tráfico entre H3 y H2 sigue el camino determinado por el IGP, que se encuentra congestionado. La Figura 13 muestra una de las pruebas de cálculo de rendimiento de red realizadas con Iperf en H3 y en la tabla 3 se muestra el resumen de las pruebas realizadas.

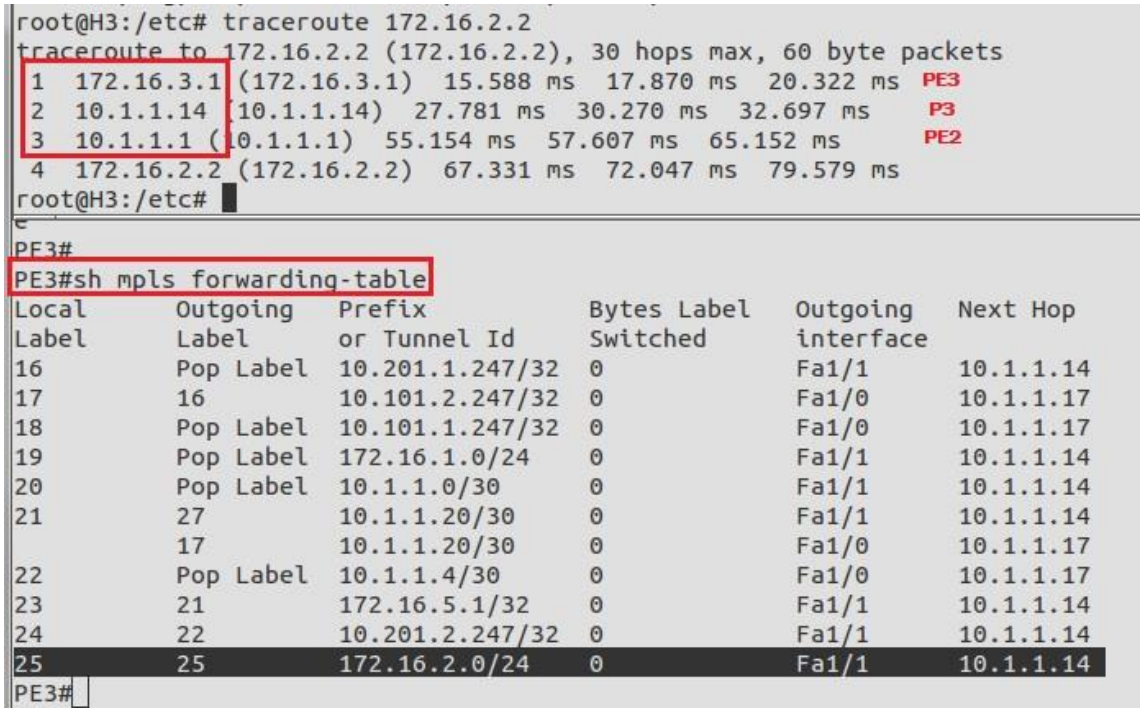

Figura 12. Experimento 1: Traza realizada desde H3 hacia H2 y tabla LFIB de PE3 sin túnel TE

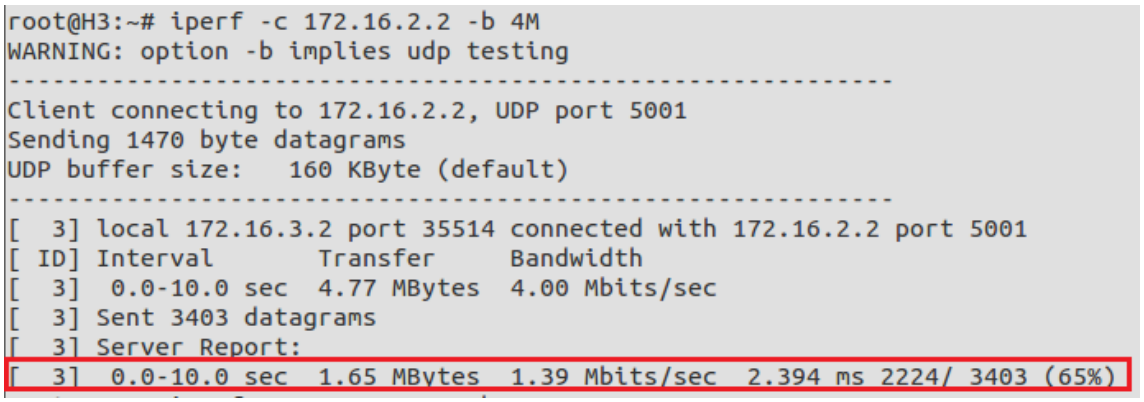

Figura 13. Experimento 1: Rendimiento medido en H3 durante congestión sin MPLS-TE

Tabla 3. Experimento 1: Pruebas Escenario 1

Sin MPLS-TE	Prueba 1	Prueba 2	Prueba 3	Prueba 4	Prueba 5	**Promedio**
Rendimiento (Mbps)	1.49	1.37	1.37	1.39	1.59	**1.44**
Jitter (ms)	1.785	3.943	4.139	2.394	19.951	**6.44**
Pérdidas (datagrama)	73%	65%	65%	65%	59%	**65%**

5.4.1.2 *Escenario 1.2: Red MPLS que experimenta congestión sin MPLS-TE.*

Para lograr que el rendimiento entre H3 y H2 mejore, se implementó un túnel TE explícito llamado "T2" en PE3 (LSR de entrada) hacia PE2 (LSR de salida). En el ios de Cisco son necesarios los siguientes comandos para la creación de un túnel TE:

```
interface Tunnel2
ip unnumbered Loopback0
tunnel mode mpls traffic-eng
tunnel destination 10.201.2.247
tunnel mpls traffic-eng priority 1 1
tunnel mpls traffic-eng bandwidth 1024
tunnel mpls traffic-eng path-option 1 explicit name BACKUP
```

Existen tres opciones para poder enrutar el tráfico a través de un túnel TE en el ios de Cisco: (1) con rutas estáticas, (2) enrutamiento basado en políticas o también llamado PBR y (3) configurar en la interface túnel el comando "tunnel mpls traffic-eng autoroute announce", se enruta todo el tráfico sin distinción hacia el LER de salida y las redes que están detrás del LER de salida. En éste escenario se utilizó una ruta estática para que solo el tráfico hacia la red 172.16.2.0/24 se envíe a través del túnel TE "T2". El túnel TE "T2" sigue el camino PE3-P1-P2-PE2. La traza y la tabla LFIB de PE3 muestran que el tráfico desde H3 hacia H2 sigue el camino establecido por "T2" como muestra la Figura 14. Y en la Figura 15 se observa una de las pruebas de rendimiento realizadas cuando el tráfico se enruta a través del túnel TE "T2". La tabla se muestra el resumen de las pruebas realizadas.

```
root@H3:/etc# traceroute 172.16.2.2
traceroute to 172.16.2.2 (172.16.2.2), 30 hops max, 60 byte packets
 1  172.16.3.1 (172.16.3.1)  22.984 ms  35.579 ms  41.654 ms   PE3
 2  10.1.1.17 (10.1.1.17)  55.088 ms  57.652 ms  60.503 ms     P1
 3  10.1.1.6 (10.1.1.6)  110.299 ms  112.725 ms  115.150 ms    P2
 4  10.1.1.22 (10.1.1.22)  92.783 ms  97.613 ms  100.025 ms
 5  172.16.2.2 (172.16.2.2)  105.133 ms  107.485 ms  117.530 ms
root@H3:/etc#
PE3#sh mpls forwarding-table
Local      Outgoing   Prefix           Bytes Label   Outgoing   Next Hop
Label      Label      or Tunnel Id     Switched      interface
16         Pop Label  10.201.1.247/32  0             Fa1/1      10.1.1.14
17         16         10.101.2.247/32  0             Fa1/0      10.1.1.17
18         Pop Label  10.101.1.247/32  0             Fa1/0      10.1.1.17
19         Pop Label  172.16.1.0/24    0             Fa1/1      10.1.1.14
20         Pop Label  10.1.1.0/30      0             Fa1/1      10.1.1.14
21         27         10.1.1.20/30     0             Fa1/1      10.1.1.14
           17         10.1.1.20/30     0             Fa1/0      10.1.1.17
22         Pop Label  10.1.1.4/30      0             Fa1/0      10.1.1.17
23         21         172.16.5.1/32    0             Fa1/1      10.1.1.14
24         22         10.201.2.247/32  0             Fa1/1      10.1.1.14
25    [T]  Pop Label  172.16.2.0/24    0             Tu2        point2point

[T]     Forwarding through a LSP tunnel.
        View additional labelling info with the 'detail' option
PE3#
```

Figura 14. Experimento1: Traza realizada desde H3 hacia H2 y tabla LFIB de PE3 utilizando túnel TE

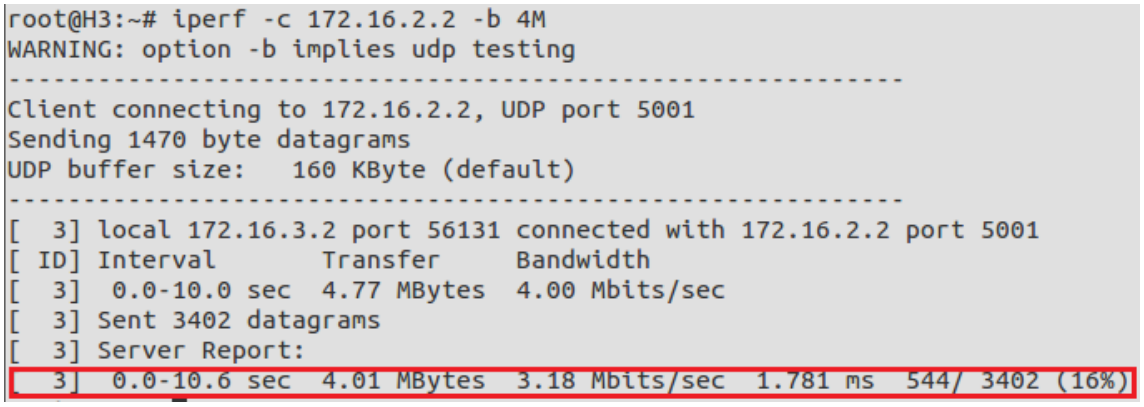

Figura 15. Experimento 1: Rendimiento medido en H3 durante congestión con MPLS-TE

Tabla 4. Experimento 1: Pruebas Escenario 2

Con MPLS-TE	Prueba 1	Prueba 2	Prueba 3	Prueba 4	Prueba 5	**Promedio**
Rendimiento (Mbps)	2.74	2.75	2.5	2.73	3.18	**2.78**
Jitter (ms)	2.874	3.741	2.038	2.069	1.781	**2.50**
Pérdidas (datagrama)	24%	23%	30%	21%	16%	**23%**

5.4.1.3 Análisis de resultados

Los resultados en la Tabla 5 demuestran que utilizando MPLS-TE se pudo optimizar el uso de los recursos de red y enrutar el tráfico entre H3 y H2 por un camino no congestionado y diferente al camino calculado por el IGP. Enrutando el tráfico a través del túnel TE "T2" permitió mejorar el rendimiento promedio entre H3 y H2 de 1.44 Mbps a 2.78 Mbps, se redujo el jitter significativamente de 6.44 ms a 2.50 ms y las pérdidas se redujeron del 65% al 23%. Las Figuras 16, 17 y 18 muestran una comparación de los resultados obtenidos en los dos escenarios de pruebas.

Tabla 5. Experimento 1: Comparación de resultados Escenario 1 y 2

Sin MPLS-TE	**Promedio**
Rendimiento (Mbps)	**1.44**
Jitter (ms)	**6.44**
Pérdidas (datagrama)	**65%**

Con MPLS-TE	**Promedio**
Rendimiento (Mbps)	**2.78**
Jitter (ms)	**2.50**
Pérdidas (datagrama)	**23%**

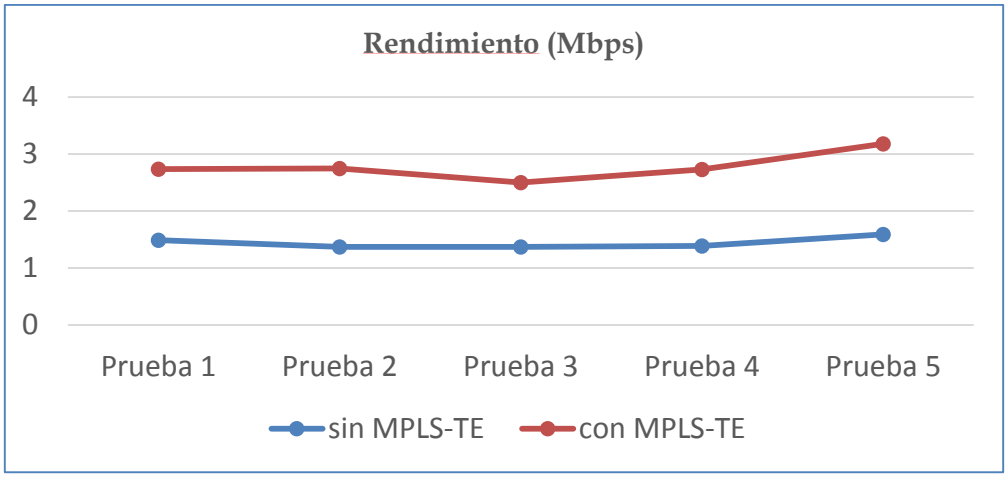

Figura 16. Experimento 1: Comparación de Rendimiento

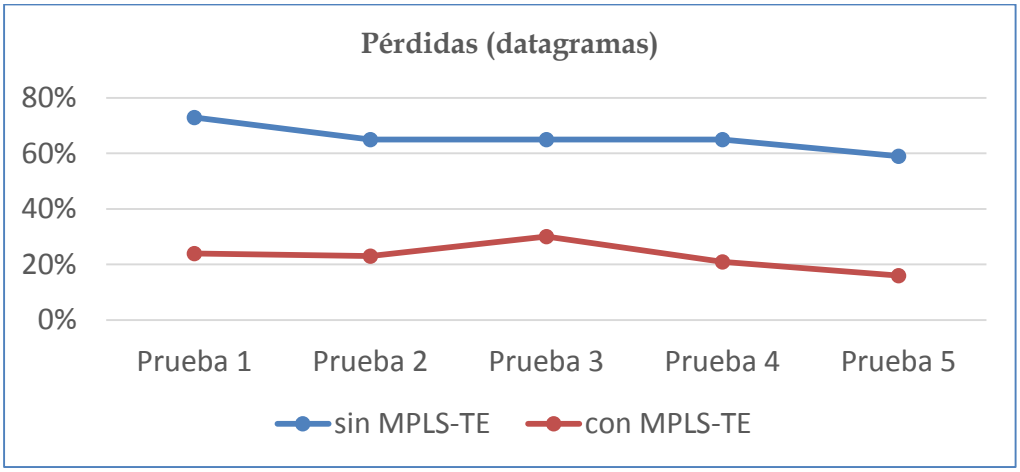

Figura 17. Experimento 1: Comparación de Pérdidas

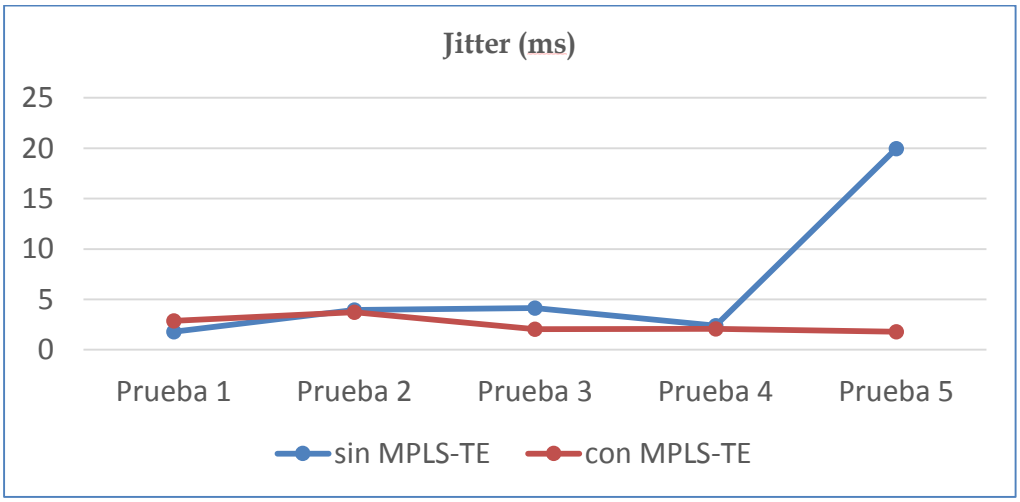

Figura 18. Experimento 1: Comparación de Jitter

5.4.2 Experimento 2: Uso de Fast Reroute

El objetivo principal de este experimento es poder observar el funcionamiento del mecanismo de restablecimiento de caminos Fast Reroute en MPLS-TE y realizar una comparación cuantitativa del tiempo de conmutación durante la convergencia de red ante un fallo en dos escenarios de red: MPLS puro y MPLS-TE con Fast Reroute. Las configuraciones de los routers se encuentran en el Anexo 1.

5.4.2.1 Escenario 2.1: Simulación de fallo de un enlace en una red MPLS

Para realizar esta prueba, se utilizó el escenario base MPLS mostrado en la Figura 9. En este escenario el IGP ha calculado automáticamente la ruta entre H2 y H3, y sigue el camino PE3-P3-PE2. Se simuló la caída del enlace entre PE2 y P3 dando de baja la interface FastEthernet1/1 del P3, esto provocó la pérdida de las sesiones OSPF y LDP que tenía establecidas con su vecino PE2, lo cual obligó al IGP a recalcular la ruta como muestra la Figura 19.

```
P3(config)#int fa1/1
P3(config-if)#shut
P3(config-if)#
*Jun 26 01:36:39.703: %OSPF-5-ADJCHG: Process 2914, Nbr 10.201.2.247 on FastEthernet1/
 from FULL to DOWN, Neighbor Down: Interface down or detached
*Jun 26 01:36:39.751: %LDP-5-NBRCHG: LDP Neighbor 10.201.2.247:0 (1) is DOWN (Interfac
 not operational)
*Jun 26 01:36:41.643: %LINK-5-CHANGED: Interface FastEthernet1/1, changed state to adm
nistratively down
*Jun 26 01:36:42.643: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet1/1,
changed state to down
          17         10.1.1.16/30     0             Fa0/1      10.1.1.21
21        Pop Label  10.1.1.12/30     0             Fa1/1      10.1.1.2
22        Pop Label  10.1.1.4/30      0             Fa0/1      10.1.1.21
23        25         172.16.4.1/32    0             Fa1/1      10.1.1.2
24        26         10.201.3.247/32  0             Fa1/1      10.1.1.2
25        27         172.16.3.0/24    0             Fa1/1      10.1.1.2
PE2#
*Jun 26 01:36:50.379: %LDP-5-NBRCHG: LDP Neighbor 10.201.1.247:0 (1) is DOWN (TCP conne
ction closed by peer)
*Jun 26 01:37:17.267: %OSPF-5-ADJCHG: Process 2914, Nbr 10.201.1.247 on FastEthernet1/1
 from FULL to DOWN, Neighbor Down: Dead timer expired
```

Figura 19. Experimento 2: Simulación de caída de enlace entre P3 y PE2 en una red MPLS

Los resultados de la prueba de ping entre los hosts H2 y H3 indican un porcentaje de pérdidas del 6% como se muestra en la Figura 20. De este resultado se puede deducir que el tiempo de convergencia de la red en MPLS puro fue de 6 segundos.

```
64 bytes from 172.16.3.2: icmp_seq=95 ttl=60 time=42.0 ms
64 bytes from 172.16.3.2: icmp_seq=96 ttl=60 time=63.0 ms
64 bytes from 172.16.3.2: icmp_seq=97 ttl=60 time=59.0 ms
64 bytes from 172.16.3.2: icmp_seq=98 ttl=60 time=53.4 ms
64 bytes from 172.16.3.2: icmp_seq=99 ttl=60 time=41.5 ms
64 bytes from 172.16.3.2: icmp_seq=100 ttl=60 time=61.0 ms

--- 172.16.3.2 ping statistics ---
100 packets transmitted, 94 received, 6% packet loss, time 99161ms
rtt min/avg/max/mdev = 15.013/38.201/83.325/13.117 ms
root@H2:~#
```

Figura 20. Experimento 2: Prueba de ping entre H2 y H3 durante convergencia de red MPLS

5.4.2.2 *Escenario 2.2: Simulación de fallo de un enlace en una red MPLS-TE con Fast Reroute*

Para la prueba de MPLS-TE con Fast Reroute, se crearon dos túneles TE en el PE3 hacia PE2 para el enrutamiento del tráfico entre H2 y H3. El túnel "T1" es el enlace principal que sigue la ruta PE3-P3-PE2 y túnel "T2" es el enlace de backup que sigue la ruta PE3-P1-P2-PE2. Debido a que los túneles TE son unidireccionales, se crearon también 2 túneles en el PE2, con el mismo esquema de enlace principal y backup para de esta manera proteger todo el tráfico entre H2 y H3. El esquema utilizado se aprecia en la Figura 21.

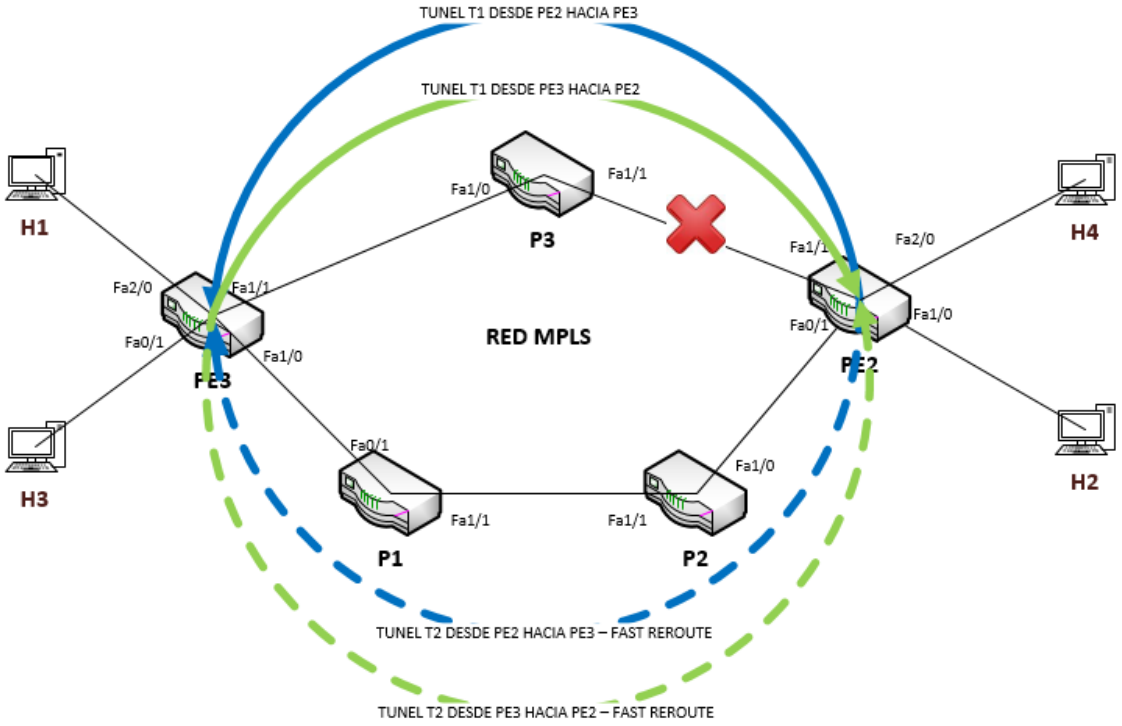

Figura 21. Diagrama Experimento 2

Se comprueba mediante una traza desde H2 hacia H3 que el tráfico está siendo enrutado a través de túnel T1 que es el enlace principal, como muestra la Figura 22.

```
root@H2:~# traceroute 172.16.3.2
traceroute to 172.16.3.2 (172.16.3.2), 30 hops max, 60 byte packets
 1  172.16.2.1 (172.16.2.1)  27.286 ms  69.851 ms  72.722 ms
 2  10.1.1.2 (10.1.1.2)  52.118 ms  93.020 ms  95.569 ms
 3  10.1.1.13 (10.1.1.13)  97.974 ms  108.517 ms  110.958 ms
 4  172.16.3.2 (172.16.3.2)  85.436 ms  87.927 ms  90.344 ms
root@H2:~#
```

Figura 22. Experimento2: Traza realizada desde H2 hacia H3 con túnel T1 en operación

Para comprobar que el esquema de protección mediante túnel T2 de backup estaba funcionando correctamente se verificó en el PE2 con el comando "sh mpls traffic-eng fast-reroute database". Éste comando permite ver información sobre los enlaces de respaldo implementados con Fast Reroute como se muestra en la Figura 23. El estatus *ready* indica que el túnel backup T2 se encuentra establecido y listo para utilizarse en caso de fallas del enlace principal túnel T1.

```
PE2#sh mpls traffic-eng fast-reroute database
P2P Headend FRR information:
Protected tunnel              In-label Out intf/label   FRR intf/label   Status
----------------------------  -------- --------------   --------------   ------
Tunnel1                       Tun hd   Fa1/1:33         Tu2:implicit-nul ready

P2P LSP midpoint frr information:
LSP identifier                In-label Out intf/label   FRR intf/label   Status
----------------------------  -------- --------------   --------------   ------

P2MP Sub-LSP FRR information:
*Sub-LSP identifier
src_lspid[subid]->dst_tunid   In-label Out intf/label   FRR intf/label   Status
----------------------------  -------- --------------   --------------   ------

* Sub-LSP identifier format: <TunSrc>_<LSP_ID>[SubgroupID]-><TunDst>_<Tun_ID>
  Note: Sub-LSP identifier may be truncated.
  Use 'detail' display for the complete key.
PE2#
```

Figura 23. Experimento 2: Base de datos de enlaces MPLS-TE Fast Reroute en PE2

Se simuló la caída del enlace entre PE2 y P3 dando de baja la interface FastEthernet1/1 del P3, como se puede observar en la Figura 24, se caen las sesiones OSPF entre P3 y PE2, sin embargo no se evidencia caída de los túneles en PE2, lo cual es el comportamiento correcto, de acuerdo a la literatura revisada en [40].

```
P3(config)#int fa1/1
P3(config-if)#shut
*Jun 26 01:10:17.011: %OSPF-4-ERRRCV: Received invalid packet: Bad LLS Checksum from 1
0.1.1.13, FastEthernet1/0
P3(config-if)#shut
P3(config-if)#
*Jun 26 01:10:27.231: %OSPF-5-ADJCHG: Process 2914, Nbr 10.201.2.247 on FastEthernet1/
1 from FULL to DOWN, Neighbor Down: Interface down or detached
*Jun 26 01:10:27.311: %LDP-5-NBRCHG: LDP Neighbor 10.201.2.247:0 (1) is DOWN (Interfac
e not operational)
*Jun 26 01:10:29.123: %LINK-5-CHANGED: Interface FastEthernet1/1, changed state to adm
inistratively down
*Jun 26 01:10:30.123: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet1/1,
 changed state to down
*Jun 26 01:11:31.987: %OSPF-4-ERRRCV: Received invalid packet: Bad LLS Checksum from 1
0.1.1.13, FastEthernet1/0
P3(config-if)#
01 10020000 FF000090 00100107
*Jun 26 01:10:11.631: %RSVP-3-MSG_1:  10020000 FF000090 00100107 0AC903F7 000000
01 EF4302F7 000C0301 0A010102
*Jun 26 01:10:11.631: %RSVP-3-MSG_2:  16000408 00080501 00007530 00080801 000000
12 00240902 00000007 05000006
*Jun 26 01:10:36.271: %LDP-5-NBRCHG: LDP Neighbor 10.201.1.247:0 (3) is DOWN (TC
P connection closed by peer)
*Jun 26 01:11:05.275: %OSPF-5-ADJCHG: Process 2914, Nbr 10.201.1.247 on FastEthe
rnet1/1 from FULL to DOWN, Neighbor Down: Dead timer expired
PE2#
```

Figura 24. Experimento 2: Simulación de caída de enlace entre P3 y PE2 en una red MPLS-TE con Fast Reroute

Una vez que se detectó la caída del enlace sobre el cual esta implementado el túnel T1, el túnel T2 entró en operación. La figura 25 muestra el estatus *active* de túnel T2 y los resultados de la prueba de ping entre los hosts H2 y H3 se muestran en la Figura 26. Este resultado de la prueba de ping permite deducir que el tiempo de convergencia fue cercano a 0 segundos.

```
PE2#sh mpls traffic-eng fast-reroute database
P2P Headend FRR information:
Protected tunnel              In-label Out intf/label   FRR intf/label   Status
----------------------------  -------- --------------   --------------   ------
Tunnel1                       Tun hd   Fa1/1:33         Tu2:implicit-nul active

P2P LSP midpoint frr information:
LSP identifier                In-label Out intf/label   FRR intf/label   Status
----------------------------  -------- --------------   --------------   ------

P2MP Sub-LSP FRR information:
*Sub-LSP identifier
src_lspid[subid]->dst_tunid   In-label Out intf/label   FRR intf/label   Status
----------------------------  -------- --------------   --------------   ------

* Sub-LSP identifier format: <TunSrc>_<LSP_ID>[SubgroupID]-><TunDst>_<Tun_ID>
  Note: Sub-LSP identifier may be truncated.
  Use 'detail' display for the complete key.
PE2#
```

Figura 25. Experimento 2: Estatus Túnel 2 después del fallo de enlace

```
64 bytes from 172.16.3.2: icmp_seq=92 ttl=60 time=39.0 ms
64 bytes from 172.16.3.2: icmp_seq=93 ttl=60 time=44.0 ms
64 bytes from 172.16.3.2: icmp_seq=94 ttl=60 time=38.5 ms
64 bytes from 172.16.3.2: icmp_seq=95 ttl=60 time=38.5 ms
64 bytes from 172.16.3.2: icmp_seq=96 ttl=60 time=26.5 ms
64 bytes from 172.16.3.2: icmp_seq=97 ttl=60 time=35.0 ms
64 bytes from 172.16.3.2: icmp_seq=98 ttl=60 time=24.5 ms
64 bytes from 172.16.3.2: icmp_seq=99 ttl=60 time=47.4 ms
64 bytes from 172.16.3.2: icmp_seq=100 ttl=60 time=29.2 ms

--- 172.16.3.2 ping statistics ---
100 packets transmitted, 100 received, 0% packet loss, time 99133ms
rtt min/avg/max/mdev = 20.472/41.511/86.511/15.330 ms
root@H2:~# traceroute 172.16.3.2
```

Figura 26. Experimento 2: Prueba de ping desde H2 hacia H3 durante fallo con MPLS Fast Reroute

Se comprobó mediante una traza desde H2 hacia H3 que el tráfico está siendo enrutado a través del túnel 2 que es el enlace de backup (Figura 27).

```
root@H2:~# traceroute 172.16.3.2
traceroute to 172.16.3.2 (172.16.3.2), 30 hops max, 60 byte packets
 1  172.16.2.1 (172.16.2.1)  73.307 ms  85.243 ms  87.856 ms
 2  10.1.1.21 (10.1.1.21)  92.794 ms *  105.113 ms
 3  10.1.1.5 (10.1.1.5)  247.544 ms  250.006 ms  252.426 ms
 4  10.1.1.18 (10.1.1.18)  142.890 ms  159.904 ms  242.277 ms
 5  172.16.3.2 (172.16.3.2)  214.607 ms  217.044 ms  234.527 ms
root@H2:~#
```

Figura 27. . Experimento2: Traza realizada desde H2 hacia H3 con "túnel 2" en operación

5.4.2.3 Análisis de resultados

En una red MPLS sin un mecanismo de protección de caminos, el tiempo de convergencia de la red será determinado por el IGP que se utilice. Como se aprecia en la Tabla 6, se obtuvo un tiempo de convergencia de 6 segundos en la red MPLS puro y al utilizar MPLS-TE con el mecanismo de Fast Reroute el tiempo de convergencia fue prácticamente inmediato. Este resultado comprueba la información encontrada en la literatura, que indica que el tiempo de convergencia utilizando el mecanismo de Fast Reroute es cercano a los 50 ms. Fast Reroute permite que una red sea más resistente en caso de fallas y es especialmente atractivo para la protección de tráfico de voz que es tan susceptible a pérdidas.

Tabla 6. Comparación de resultados Experimento 2

Tipo de Red	Pérdidas (paquetes)	Tiempo/Convergencia (seg)
Red MPLS	6%	6
Red MPLS-TE FRR	0%	0

5.4.3 Experimento 3: Balanceo de carga en enlaces con diferente costo

El objetivo de esta prueba es poder comprobar como MPLS-TE realiza balanceo de carga en enlaces con diferente costo. En este experimento se medirá el rendimiento, jitter y porcentaje de pérdida de datagramas en dos escenarios de red:

5.4.3.1 Escenario 3.1: Red MPLS que experimenta congestión, sin balanceo de carga.

En la red mostrada en la Figura 9 se ha simulado congestión utilizando Iperf para enviar 4Mbps de tráfico entre H1 y H4 y congestionar el camino calculado por el IGP que es PE3-P3-PE2. El tráfico desde H3 hacia H2 sigue el camino calculado por el IGP. Los resultados de rendimiento de red se muestran en la Tabla 7.

Tabla 7. Experimento 3: Resultados de rendimiento sin balanceo de carga

Sin Balanceo de carga	Prueba 1	Prueba 2	Prueba 3	Prueba 4	Prueba 5	**Promedio**
Rendimiento (Mbps)	1.66	1.59	1.78	1.63	1.66	**1.66**
Jitter (ms)	2.484	20.692	15.147	18.15	5.267	**12.35**
Pérdidas (datagrama)	58%	56%	53%	58%	58%	**57%**

5.4.3.2 Escenario 3.2: Red MPLS que experimenta congestión, con balanceo de carga.

En este escenario se crearon dos túneles TE explícitos para el enrutamiento del tráfico entre H2 y H3, túnel "T1" cuya ruta es PE3-P3-PE2 y costo calculado por el IGP es 2, y un segundo túnel "T2" cuya ruta está formada por PE3-P1-P2-PE2, con costo 3 (Figura 28). A T1 fue configurado con atributo de ancho de banda de 512 Kbps y T2 fue configurado con atributo de ancho de banda de 1024 Kbps; por tanto la relación para el balanceo de tráfico será 2:1. Esto significa que por cada 2 paquetes enviados a través del túnel T2, se envía 1 paquete a través de túnel T1. Las configuraciones de los routers se encuentran en el Anexo 3.

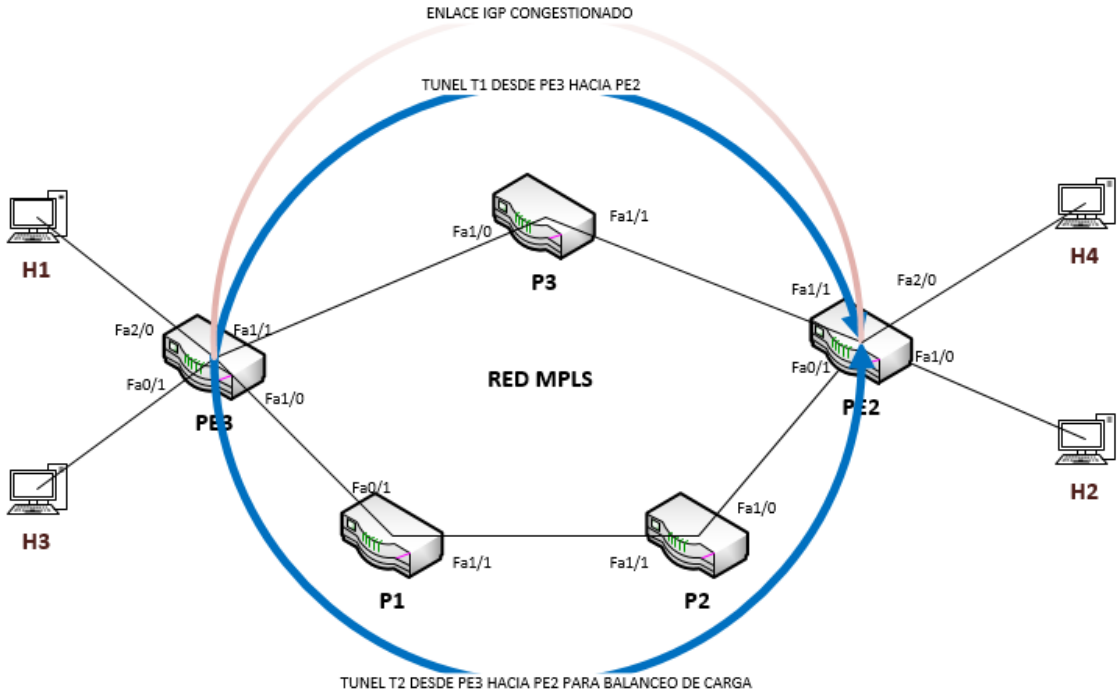

Figura 28. Diagrama Experimento 3

Como vemos en la Figura 29, desde el PE3 existen dos caminos para llegar a la red 172.16.2.0/24, donde reside el host H2. La línea "traffic share count 1" para el túnel T2 y "traffic share count 2" para el túnel T1, indica que por cada paquete enviado por el túnel T2, se envían dos paquetes por el túnel T1. Esta información se puede verificar también en la tabla *cef* del router PE3 con el comando "show ip cef 172.16.2.0 internal", como se muestra en la Figura 30. Si se cuenta el número de ocurrencias del Tunnel2 se confirmara que es el doble de ocurrencias del Tunnel 1 lo cual indica que la relación para envío de tráfico hacia la red 172.16.2.0/24 es 2:1. Los resultados de rendimiento de red se muestran en la Tabla 8.

```
PE3#sh ip route 172.16.2.2
Routing entry for 172.16.2.0/24
  Known via "static", distance 1, metric 0 (connected)
  Routing Descriptor Blocks:
    directly connected, via Tunnel2
      Route metric is 0, traffic share count is 2
  * directly connected, via Tunnel1
      Route metric is 0, traffic share count is 1
PE3#
```

Figura 29. Experimento3: Balanceo de carga en enlaces con diferente costo

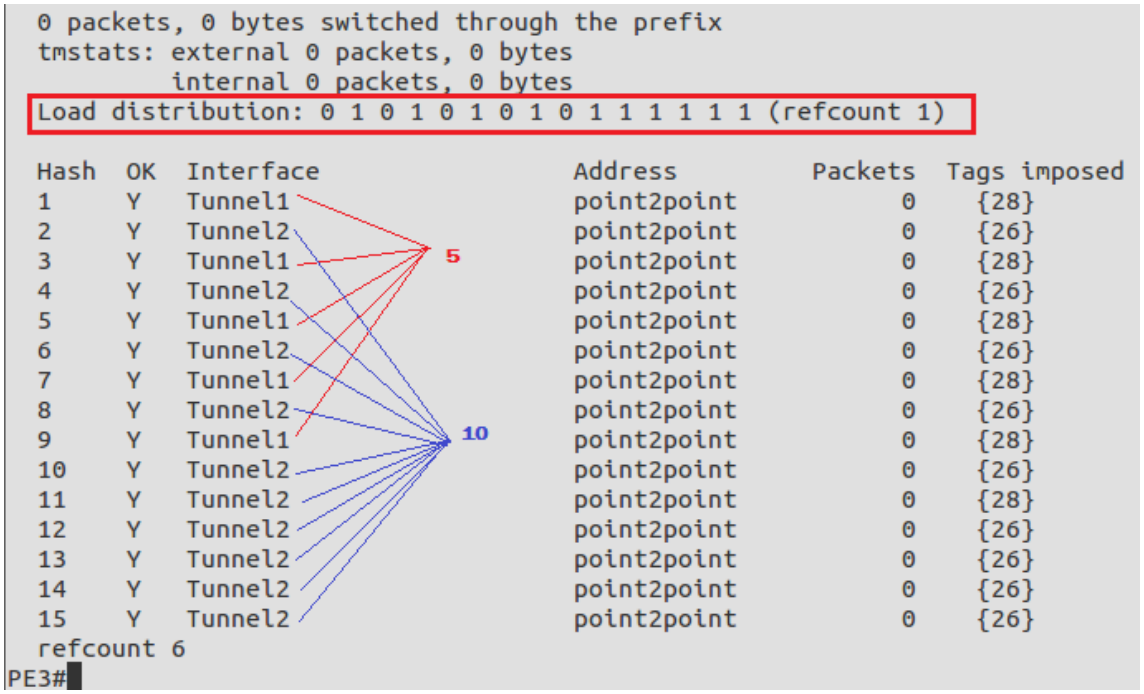

Figura 30. Experimento 3: Tabla CEF del router PE3

Tabla 8. Experimento 3: Resultados de rendimiento con balanceo de carga

Con Balanceo de carga	Prueba 1	Prueba 2	Prueba 3	Prueba 4	Prueba 5	**Promedio**
Rendimiento (Mbps)	3.34	3.22	3.29	3.56	3.02	**3.29**
Jitter (ms)	0.614	0.402	9.975	1.55	3.674	**3.24**
Pérdidas (datagrama)	12%	14%	13%	6%	20%	**13%**

5.4.3.3 Análisis de resultados

Comparando los valores promedio de rendimiento, jitter y pérdida de paquetes mostrados en las Tablas 7 y 8 se puede observar que todos los parámetros analizados mejoraron considerablemente; el rendimiento de 1.66 Mbps a 3.29 Mbps, el jitter de 12.35 ms a 3.34 ms y las pérdidas del 57% al 13%. La utilización del balanceo de carga a través de túneles MPLS-TE permite aprovechar los recursos disponibles en la red y mejorar las prestaciones especialmente en casos de congestión. Una gran ventaja en comparación con las redes IP donde solo es posible hacer balanceo de carga en enlaces con igual costo. Las Figuras 31, 32 y 33 muestran una comparación de los resultados obtenidos en los dos escenarios de pruebas.

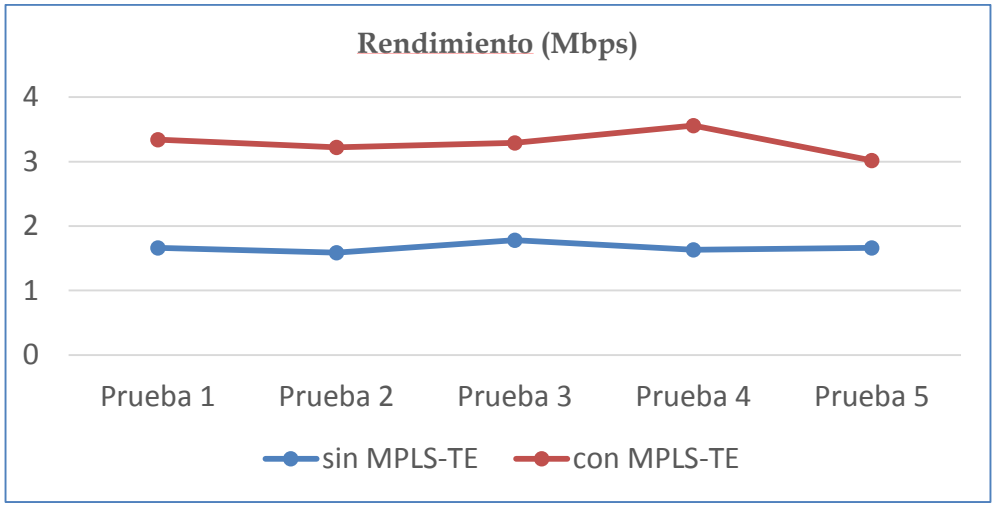

Figura 31. Experimento 3: Comparación de Rendimiento

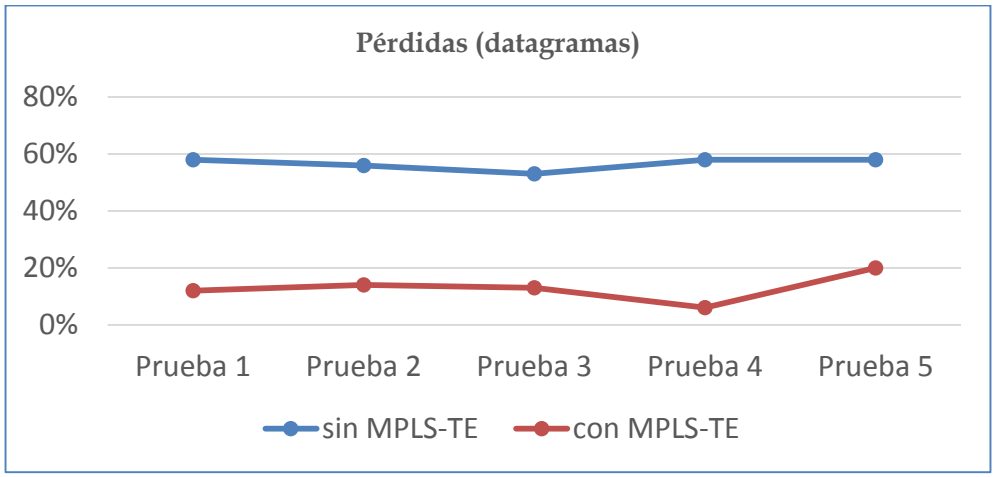

Figura 32. Experimento 3: Comparación de Pérdidas

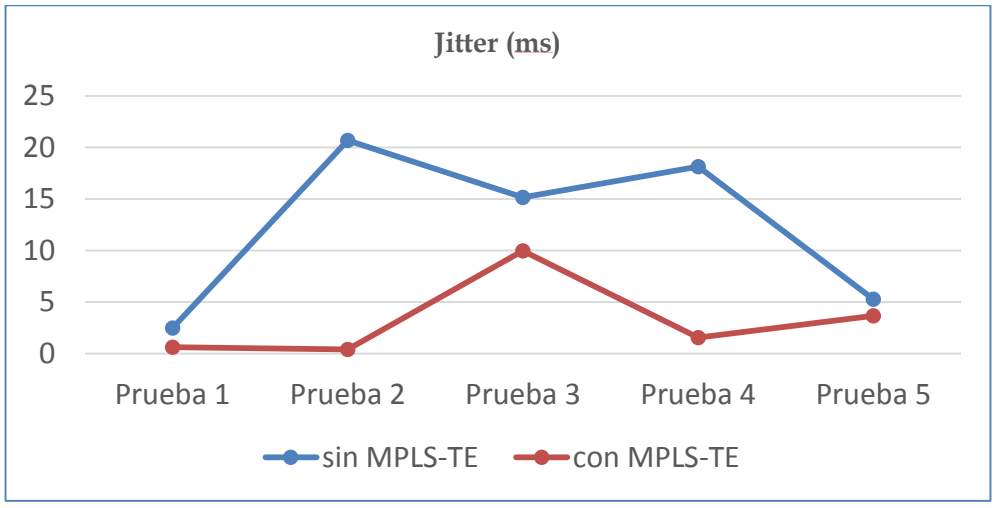

Figura 33. Experimento 3: Comparación de Jitter

5.4.4 Experimento 4: Uso de DS-TE

Se tiene un servidor de VoIP (H2) y se desea dar prioridad al tráfico de voz sobre el tráfico de datos. Se considera como tráfico de voz el tráfico enviado desde la red 172.16.3.0/24 y 172.16.1.0/24 (H1 y H3) hacia la IP del servidor VoIP (172.16.2.2). Cualquier otro tipo de tráfico se considera tráfico best effort o de máximo esfuerzo. Se utilizó EXP 5 para marcar el tráfico de voz y EXP 0 para el tráfico best effort.

Para verificar que el router estaba marcando el tráfico de voz con el campo EXP correcto, se utilizó Wireshark, y se comprobó que el PE3 estaba colocando las marcas con el campo EXP 5 que le corresponde al tráfico de voz y EXP 0 al tráfico best effort. Además se pudo observar que la etiqueta con la que eran enviados los paquetes, determinada por "Label 21". Las figuras 34 y 35 muestran los resultados del análisis de paquetes de Wireshark.

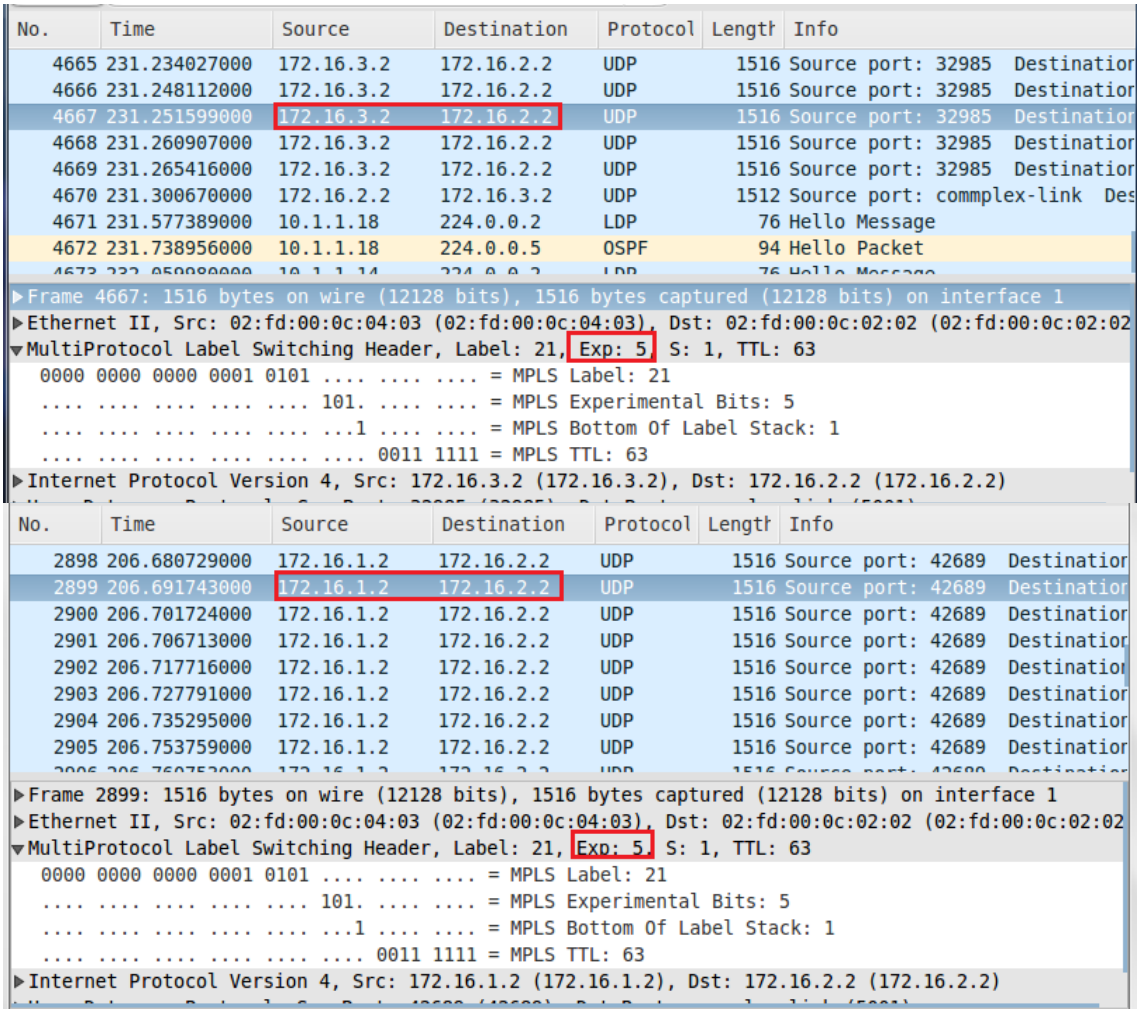

Figura 34. Experimento 4: Paquetes marcados con EXP 5 (tráfico de voz) sin MPLS-TE

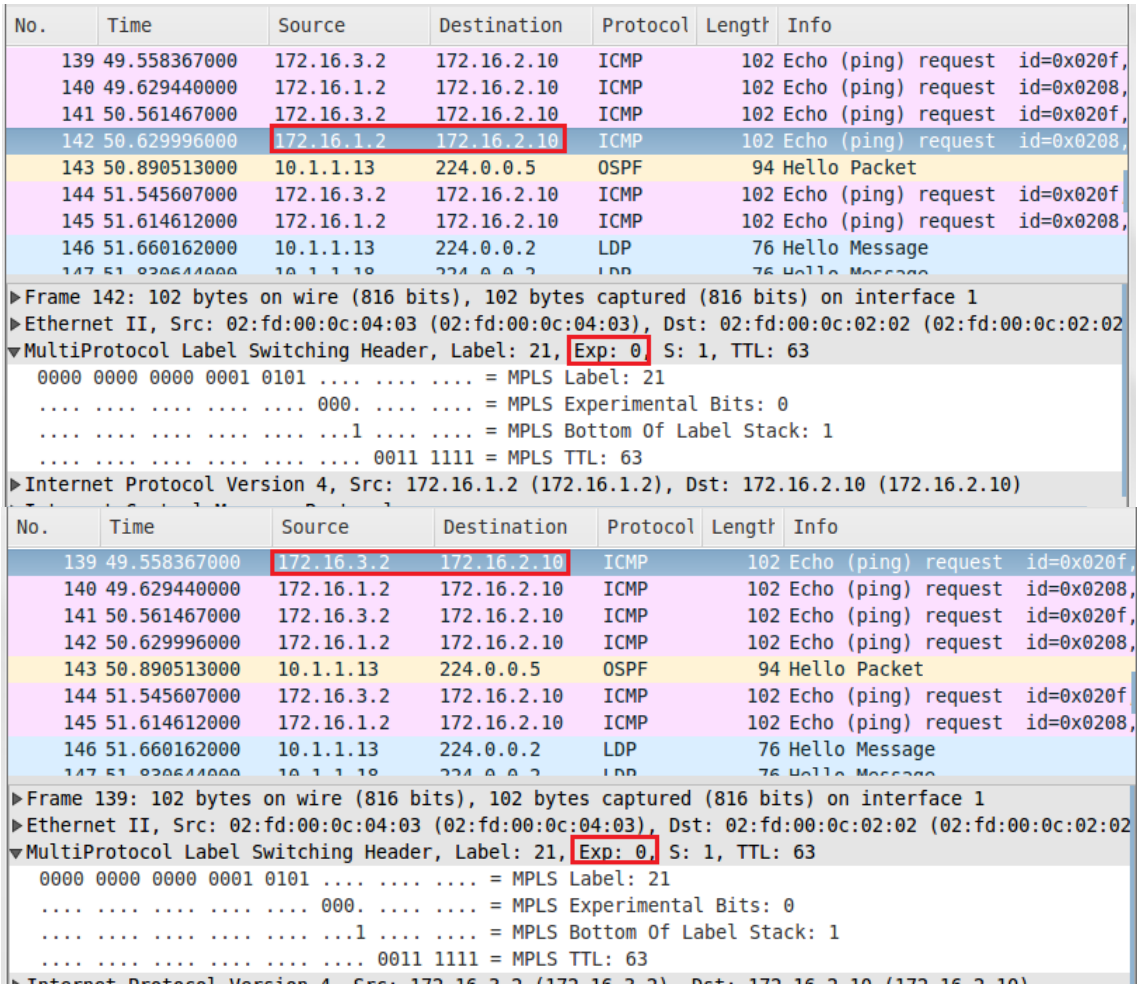

Figura 35. Paquetes marcados con EXP 0 (tráfico best effort)

Adicionalmente se verifico que la política de QoS aplicada estuviese funcionando en el router, es decir, que los paquetes de voz estuvieran encolados en la cola de prioridad correspondiente. En los routers Cisco se puede monitorear los resultados de la aplicación de una política de QoS con el comando "*show policy-map interface*". Los resultados se muestran en la figura 36.

```
PE3#sh policy-map interface fa1/1
 FastEthernet1/1

  Service-policy output: voz-y-datos

    queue stats for all priority classes:
      Queueing
      queue limit 64 packets
      (queue depth/total drops/no-buffer drops) 0/0/0
      (pkts output/bytes output) 4242/6430872

    Class-map: exp-5-voz (match-all)
      4242 packets, 6430872 bytes
      30 second offered rate 0000 bps, drop rate 0000 bps
      Match: mpls experimental topmost 5
      Priority: 40% (40000 kbps), burst bytes 1000000, b/w exceed drops: 0


    Class-map: class-default (match-any)
      1366 packets, 138476 bytes
      30 second offered rate 0000 bps, drop rate 0000 bps
      Match: any
      Queueing
      queue limit 64 packets
      (queue depth/total drops/no-buffer drops) 0/0/0
      (pkts output/bytes output) 1366/138476
      bandwidth 60% (60000 kbps)
PE3#
```

Figura 36. Experimento 4: Resultados de la política de QoS aplicada en la Interface FastEthernet1/1 del PE3

Para poder verificar los beneficios de MPLS-TE, se realizara una comparación cuantitativa de los valores de rendimiento, jitter, y perdidas, en una red MPLS con QoS y una red MPLS-TE con QoS.

5.4.4.1 Escenario 4.1: Red MPLS con políticas de QoS

En este escenario se ha utilizado el diagrama base que se presenta en la Figura 9, se utilizó Iperf para enviar 4Mbps de tráfico entre H1 y H4 y congestionar el camino calculado por el IGP que es PE3-P3-PE2. En la Tabla 9 se presentan los resultados de las pruebas.

Tabla 9. Experimento 4: Pruebas en red MPLS con QoS sin Ingeniería de tráfico

	Prueba 1	Prueba 2	Prueba 3	Prueba 4	Prueba 5	**Promedio**
Rendimiento (Mbps)	1.23	1.21	1.24	1.22	1.22	**1.22**
Jitter (ms)	4.568	7.605	4.969	9.945	7.948	**7.01**
Pérdidas (datagrama)	0%	0%	0%	0%	0%	**0%**

5.4.4.2 Escenario 4.2: Red MPLS con políticas de QoS con ingeniería de tráfico

En este escenario se utilizó Iperf para enviar 4Mbps de tráfico entre H1 y H4 y congestionar el camino calculado por el IGP que es PE3-P3-PE2. Se configuro un túnel TE explícito "T1", con prioridad 0 (la más alta). El túnel "T1" sigue el camino PE3-P3-PE2 con ancho de banda reservado de 1.229 Mbps para dar calidad de servicio al tráfico de voz como se muestra en la Figura 37. Los comandos utilizados para la configuración del túnel son los siguientes:

```
interface Tunnel1
 ip unnumbered Loopback0
 tunnel mode mpls traffic-eng
 tunnel destination 10.201.2.247
 tunnel mpls traffic-eng priority 0 0
 tunnel mpls traffic-eng bandwidth sub-pool 1229
 tunnel mpls traffic-eng path-option 1 explicit name PRINCIPAL
```

Dado que la política de calidad de servicio limita el ancho de banda de voz a 1.229 Mbps, se utilizó este valor como parámetro para medir el rendimiento de la red.

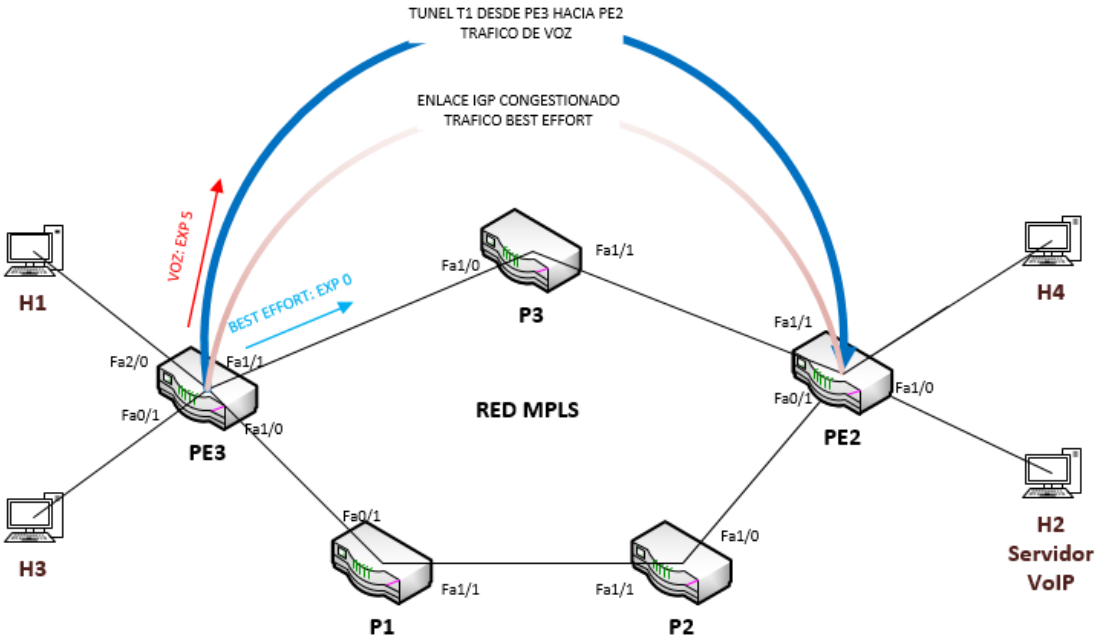

Figura 37. Diagrama Experimento 4

En la Tabla 10 se presentan los resultados de las pruebas, y en la Figura 38 se observan los resultados de Wireshark donde se evidencia que los paquetes de voz

están marcado con EXP 5 y son enviados con la etiqueta 16 que corresponde al túnel T1 como lo demuestra la Figura 39.

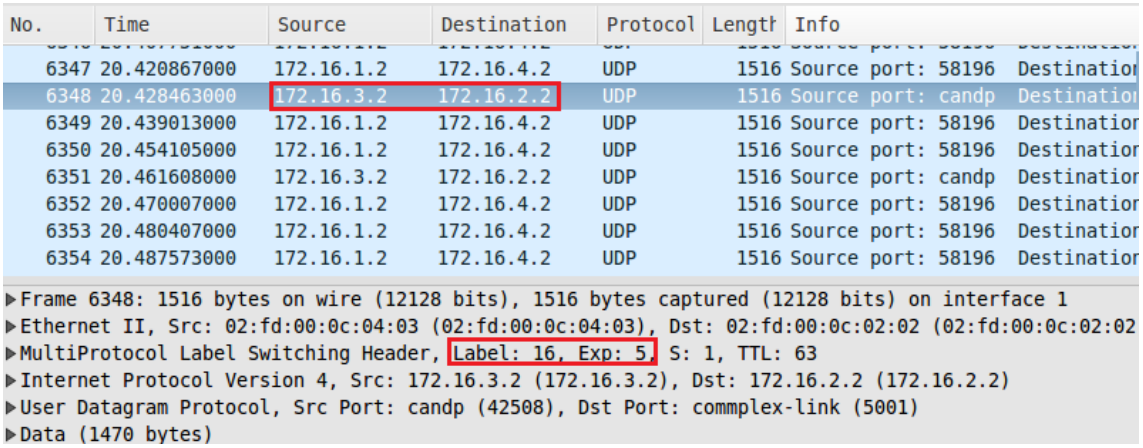

Figura 38. Experimento 4: Paquetes de voz marcados con EXP 5 y enviados con label 16

```
PE3#sh ip cef 172.16.2.0 internal
172.16.2.0/24, epoch 0, flags attached, RIB[S], refcount 5, per-destination shar
ing
  sources: RIB, LTE
  feature space:
   IPRM: 0x00048004
   LFD: 172.16.2.0/24 1 local label
   local label info: global/21
        contains path extension list
        disposition chain 0x683215F0
        label switch chain 0x68321650
  ifnums:
   Tunnel1(11)
  path 68CC00F4, path list 6829A150, share 1/1, type attached prefix, for IPv4
    MPLS short path extensions: MOI flags = 0x1 label implicit-null
  attached to Tunnel1, adjacency IP midchain out of Tunnel1 68CAFAC0
  output chain: IP midchain out of Tunnel1 68CAFAC0 label 16 TAG adj out of Fast
Ethernet1/1, addr 10.1.1.14 676393C0
```

Figura 39. Experimento 4: Label colocada a los paquetes que se envían por el túnel T1

Tabla 10. Experimento 4: Pruebas en red MPLS con QoS con Ingeniería de tráfico

	Prueba 1	Prueba 2	Prueba 3	Prueba 4	Prueba 5	**Promedio**
Rendimiento (Mbps)	1.23	1.23	1.23	1.24	1.23	**1.23**
Jitter (ms)	1.961	3.635	4.640	3.270	4.512	**3.60**
Pérdidas (datagrama)	0%	0%	0%	0%	0%	**0%**

5.4.4.3 Análisis de resultados

La política de QoS que se aplicó para dar prioridad a los paquetes de voz funcionó correctamente, tanto en una red MPLS como en la red MPLS-TE, ya que el rendimiento promedio obtenido fue de 1.23 Mbps en ambos escenarios con congestión, sin pérdida de paquetes. Sin embargo el valor promedio de jitter obtenido en la red MPLS-TE fue casi la mitad del valor de jitter de una red MPLS en la que no se utiliza Ingeniería de tráfico. Esta diferencia es muy importante especialmente cuando se trata de tráfico de voz, que es muy susceptible a pérdidas y jitter. MPLS-TE supone un mecanismo efectivo para tráfico de alta prioridad. Además se consigue dar calidad de servicio de punta a punta que resulta muy complicado conseguir en otro tipo de redes. Las Figuras 40, 41 y 42 muestran una comparación de los resultados obtenidos en los dos escenarios de pruebas

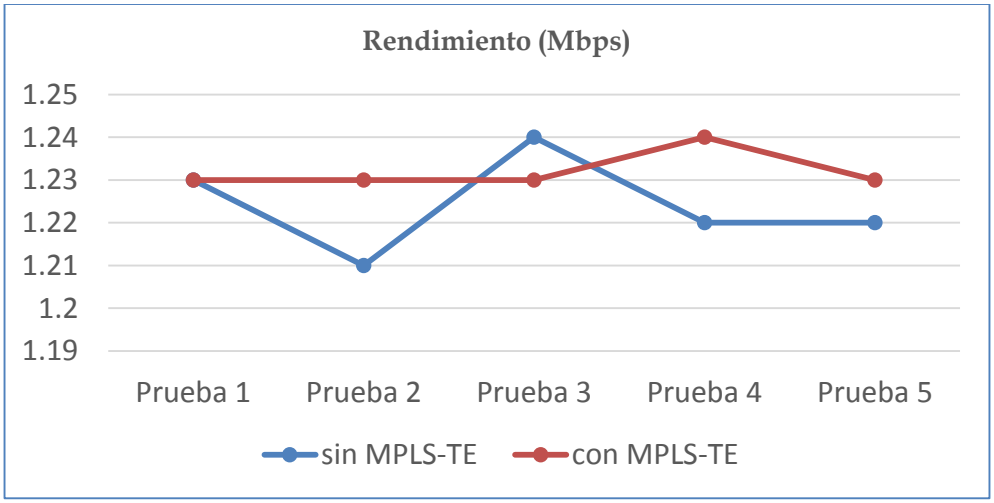

Figura 40. Experimento 4: Comparación de Rendimiento

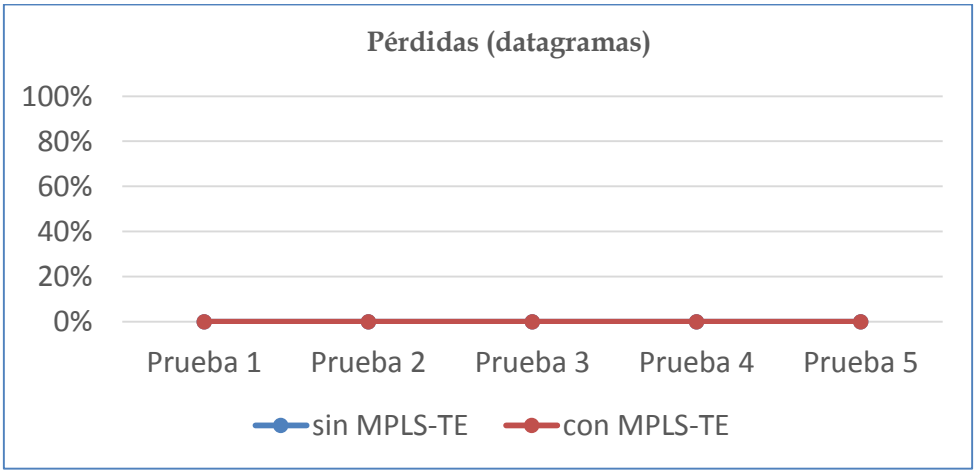

Figura 41. Experimento 4: Comparación de Pérdidas

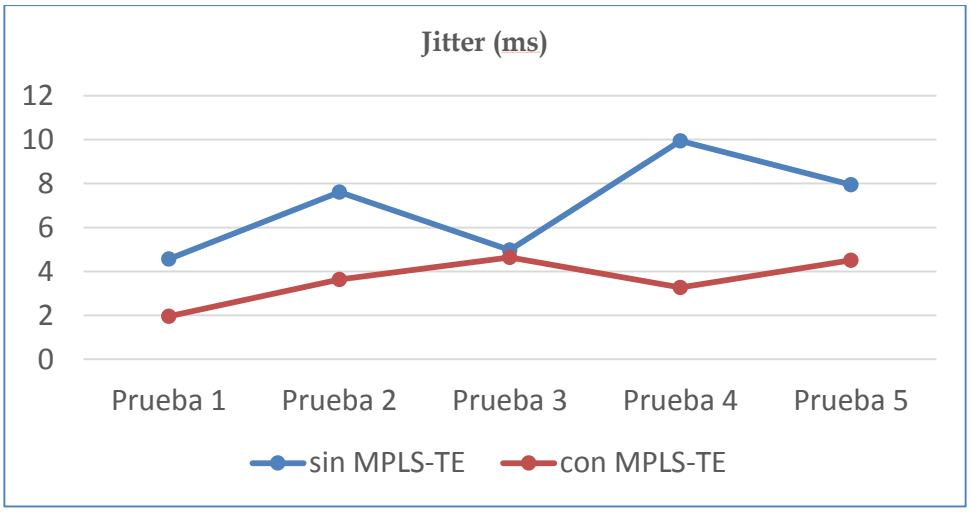

Figura 42. Experimento 4: Comparación de Jitter

6 Conclusiones

La Ingeniería de Trafico en redes MPLS ofrece importantes herramientas de optimización de red que pueden ser utilizadas en situaciones de congestión, mediante el mecanismo de túneles TE que permiten enrutar el tráfico en caminos diferentes al definido por el IGP, obteniendo así mejores niveles de rendimiento de red y optimización en el uso de los recursos disponibles, especialmente el ancho de banda, que constituye uno de los recursos más críticos para los proveedores de servicio.

Además de facilitar el manejo de tráfico en casos de congestión de red, MPLS-TE ofrece mecanismos de recuperación ante fallos como Fast Reroute y mecanismos para dar calidad como lo es DS-TE, que combina las ventajas de MPLS-TE y Diffserv. La capacidad de poder combinar estos dos mecanismos permitiría ofrecer niveles de calidad de servicio muy elevados a las aplicaciones que son más sensibles al retardo y perdidas como por ejemplo la voz sobre IP. MPLS-TE permite conseguir el despliegue de un esquema de calidad de servicio de punta a punta, a través de una configuración sencilla entre los nodos involucrados, a diferencia de otras soluciones QoS que requieren complejas configuraciones nodo a nodo.

Mediante los cuatro casos prácticos de uso de mecanismos de Ingeniería de Trafico en redes MPLS se ha logrado conseguir una mejor comprensión de la tecnología y conocer los mecanismos *online* que la tecnología facilita para poder gestionar eficientemente los recursos en una red MPLS intradominio, comprendiendo las ventajas que se consiguen a través del uso de MPLS-TE en comparación con el uso de una red MPLS puro.

Como experiencia adquirida durante el desarrollo de este trabajo es importante mencionar la habilidad en el uso de la herramienta de simulación VNX, ya que no tenía conocimientos previos de su uso. Y como resultado del desarrollo de la parte práctica de esta memoria, surgió una oportunidad de mejora para la herramienta, que es la integración de una funcionalidad que permita al usuario escoger el mtu de los enlaces emulados en su escenario de red.

Como trabajo futuro sería muy interesante poder estudiar también mediante casos prácticos los mecanismos de Ingeniería de Trafico offline, que son mecanismos que permiten hacer cálculos globales de red e implementar una solución de red tomada como un conjunto, permitiendo optimizar el uso de los recursos de red globalmente.

Bibliografía

[1] F. Ahmed and D. I. Zafar, "Analysis of traffic engineering parameters while using multi-protocol label switching (MPLS) and traditional IP networks," *Asian Transactions on Engineering (ATE ISSN: 2221-4267) Volume,*vol. 1, 2011.

[2] V. Alwayn, *Advanced MPLS Design and Implementation.* Old Tappan: Cisco Press, 2001.

[3] L. Andersson, P. Doolan, N. Feldman, A. Fredette and B. Thomas, "LDP specification," *Work in Progress,* 2001.

[4] L. Andersson and G. Swallow, *The Multiprotocol Label Switching (MPLS) Working Group Decision on MPLS Signaling Protocols,* 2003.

[5] M. N. ASLAM and Y. AZIZ, "Traffic Engineering with Multi-Protocol Label Switching," 2008.

[6] D. O. Awduche and J. Agogbua, "Requirements for traffic engineering over MPLS," 1999.

[7] D. O. Awduche and B. Jabbari, "Internet traffic engineering using multi-protocol label switching (MPLS)," *Computer Networks,* vol. 40, pp. 111-129, 2002.

[8] D. Awduche, L. Berger, D. Gan, T. Li, V. Srinivasan and G. Swallow, *RSVP-TE: Extensions to RSVP for LSP Tunnels,* 2001.

[9] D. Awduche, A. Chiu, A. Elwalid, I. Widjaja and X. Xiao, *Overview and Principles of Internet Traffic Engineering,* 2002.

[10] S. Balon, *Contributions to Traffic Engineering and Resilience in Computer Networks,* 2008.

[11] T. Bates, R. Chandra, D. Katz and Y. Rekhter, *Multiprotocol Extensions for BGP-4,* 2007.

[12] P. Belzarena, "Ingeniería de tráfico en línea en redes MPLS aplicando la teoría de grandes desviaciones," *Universidad De La República, Uruguay,* 2003.

[13] E. Calle, A. Urra, J. L. Marzo, G. Kuo and H. Guo, "Minimum interference routing with fast protection," *Communications Magazine, IEEE,* vol. 44, pp. 104-111, 2006.

[14] S. Dasgupta, J. C. de Oliveira and J. Vasseur, "A performance study of IP and MPLS traffic engineering techniques under traffic variations," in *Global Telecommunications Conference, 2007. GLOBECOM'07. IEEE,* pp. 2757-2762, 2007.

[15] A. Delfino, S. Rivero and M. SanMartín, "Ingeniería de tráfico en redes MPLS," in 2006.

[16] (Jun 29,2015). *Virtual Networks over linuX*. Available: vnx.dit.upm.es.

[17] A. Elwalid, C. Jin, S. Low and I. Widjaja, "MATE: MPLS adaptive traffic engineering," in *INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, pp. 1300-1309*,* 2001.

[18] G. B. Figueiredo, N. L. Da Fonseca and J. A. Monteiro, "A minimum interference routing algorithm with reduced computational complexity," *Computer Networks,* vol. 50, pp. 1710-1732, 2006.

[19] M. K. Girish, B. Zhou and J. Q. Hu, "Formulation of the traffic engineering problems in MPLS based IP networks," in *Computers and Communications, 2000. Proceedings. ISCC 2000. Fifth IEEE Symposium on*, pp. 214-219*,* 2000.

[20] D. Grossman, "New terminology and clarifications for diffserv," 2002.

[21] H. Hodzic and S. Zoric, "Traffic engineering with constraint based routing in MPLS networks," in *ELMAR, 2008. 50th International Symposium*, pp. 269-272*,* 2008.

[22] M. Huerta, X. Hesselbach, R. Fabregat, J. Padilla, O. Ravelo and R. Clotet, "Reducción de congestión mediante técnicas de optimización de flujos en redes MPLS," *Latin America Transactions,* 2007.

[23] L. Hundessa and J. Domingo-Pascual, "Reliable and fast rerouting mechanism for a protected label switched path," in *Global Telecommunications Conference, 2002. GLOBECOM'02. IEEE*, pp. 1608-1612*,* 2002.

[24] (Jun 29, 2015). *Iperf*. Available: https://iperf.fr/.

[25] S. Kandula, D. Katabi, B. Davie and A. Charny, "Walking the tightrope: Responsive yet stable traffic engineering," in *ACM SIGCOMM Computer Communication Review*, pp. 253-264*,* 2005.

[26] K. Kar, M. Kodialam and T. Lakshman, "Minimum interference routing of bandwidth guaranteed tunnels with MPLS traffic engineering applications," *Selected Areas in Communications, IEEE Journal on,* vol. 18, pp. 2566-2579, 2000.

[27] K. Kar, M. Kodialam and T. Lakshman, "MPLS traffic engineering using enhanced minimum interference routing: An approach based on lexicographic max-flow," in *Quality of Service. IWQOS. 2000 Eighth International Workshop on,* 2000, pp. 105-114*, 2000*.

[28] D. Katz, K. Kompella and D. Yeung, *Traffic Engineering (TE) Extensions to OSPF Version 2,* 2003.

[29] W. Lai and F. L. Faucheur, "Requirements for support of differentiated services-aware MPLS traffic engineering," 2003.

[30] T. Li and H. Smit, *IS-IS Extensions for Traffic Engineering,* 2008.

[31] E. K. Lua, J. Crowcroft, M. Pias, R. Sharma and S. Lim, "A survey and comparison of peer-to-peer overlay network schemes," *Communications Surveys & Tutorials, IEEE,* vol. 7, pp. 72-93, 2005.

[32] K. Manayya, "Constrained shortest path first," 2010.

[33] R. J. Mateo, C. P. Paniagua, A. G. Cervero, J. L. G. Sánchez and F. J. R. Pérez, "Integración de MPLS y DiffServ en una Arquitectura para la Provisión de QoSc," 2004.

[34] Y. D. Meisel, R. Fabregat, J. L. Marzo and E. Calle, "Extensión de los métodos Hop-by-Hop, CR-LDP y RSVP-TE para Multicast IP sobre MPLS," .

[35] S. Q. Mirkar and V. T. Raisinghani, "Multi protocol label switching recovery mechanism," in *Signal Propagation and Computer Technology (ICSPCT), 2014 International Conference on*, pp. 492-497*,* 2014.

[36] A. S. G. Mittal, *QoS and Traffic Engineering: MPLS, DiffServ and Constraint Based Routing,* 2000.

[37] S. Naveed and S. V. Kumar, "MPLS Traffic Engineering–Fast Reroute," .

[38] S. d. Oliveira Guerra, *Una Propuesta De Arquitectura MPLS/DiffServ Para Proveer Mecanismos De Calidad De Servicio (QOS) En El Transporte De La Telefonía IP,* 2004.

[39] B. J. Oommen, S. Misra and O. Granmo, "Routing bandwidth-guaranteed paths in MPLS traffic engineering: a multiple race track learning approach," *Computers, IEEE Transactions on,* vol. 56, pp. 959-976, 2007.

[40] E. D. Osborne and A. Simha, *Traffic Engineering with MPLS.* Cisco Press, 2002.

[41] P. Pan, G. Swallow and A. Atlas, *Fast Reroute Extensions to RSVP-TE for LSP Tunnels,* 2005.

[42] M. Pištek and M. Medvecký, "Class-based constraint-based routing with implemented fuzzy logic in MPLS-TE networks," *Journal of Computer Networks and Communications,* vol. 2014, 2014.

[43] M. K. Porwal, A. Yadav and S. Charhate, "Multimedia traffic analysis of MPLS and non MPLS network," *International Journal of Computer Science and Applications,* vol. 1, 2008.

[44] C. Press, "MPLS fundamentals," 2007.

[45] R. K. Singh, N. S. Chaudhari and K. Saxena, "Load balancing in IP/MPLS networks: A survey," 2012.

[46] K. Sinha and S. Patek, "Opiate: Optimization integrated adaptive traffic engineering," *2002-11-12).Http://www.Sys.Virginia.edu/techreps/2002/sie-020001.Pdf,* 2002.

[47] A. R. Sulaiman and O. K. S. Alhafidh, "Performance Analysis of Multimedia Traffic over MPLS Communication Networks with Traffic Engineering," *International Journal of Computer Networks & Communications Security,*vol. 2, 2014.

[48] B. Wang, X. Su and C. P. Chen, "A new bandwidth guaranteed routing algorithm for MPLS traffic engineering," in *Communications, 2002. ICC 2002. IEEE International Conference on*, pp. 1001-1005*,* 2002.

[49] (Jun 29,2015). *Wireshark*. Available: www.wireshark.org.

[50] J. Wroclawski, "The use of RSVP with IETF integrated services," 1997.

[51] L. Wu and T. Worster, "Constraint-Based LSP Setup using LDP Status of this Memo This document specifies an Internet standards track protocol for the Internet community, and requests discussion and suggestions for improvements. Please refer to the current edition of the" Internet," 2002.

[52] D. Zhang and D. Ionescu, "QoS performance analysis in deployment of DiffServ-aware MPLS traffic engineering," in *Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing, 2007. SNPD 2007. Eighth ACIS International Conference on*, pp. 963-967*,* 2007.

Anexos

Anexo 1. Configuraciones de los routers Experimento 1

Router P1

```
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname P1
!
boot-start-marker
boot-end-marker
!
enable password xxxx
!
no aaa new-model
!
resource policy
!
ip subnet-zero
ip cef
!
mpls traffic-eng tunnels
mpls traffic-eng tunnels reoptimize timers frequency 60
!
no ip dhcp use vrf connected
!
no ip domain lookup
no ip ips deny-action ips-interface
!
ipv6 unicast-routing
!
username root password 0 xxxx
!
interface Loopback0
 ip address 10.101.1.247 255.255.255.255
!
interface FastEthernet0/0
 description mgm_interface
 mac-address 02fd.000c.0000
 ip address 10.250.0.202 255.255.255.252
 duplex auto
 speed auto
!
interface FastEthernet0/1
 description PE3_fa1/0
 mac-address 02fd.000c.0001
 ip address 10.1.1.17 255.255.255.252
 duplex auto
```

```
 speed auto
 mpls ip
 mpls mtu 1700
 ip rsvp bandwidth 2048 2048
 mpls traffic-eng tunnels
!
interface FastEthernet1/1
 description P2_fa1/1
 mac-address 02fd.000c.0002
 ip address 10.1.1.5 255.255.255.252
 duplex auto
 speed auto
 mpls ip
mpls mtu 1700
ip rsvp bandwidth 2048 2048
mpls traffic-eng tunnels
!
router ospf 2914
 router-id 10.101.1.247
 log-adjacency-changes
  mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
 network 10.1.1.4 0.0.0.3 area 0
 network 10.1.1.16 0.0.0.3 area 0
 network 10.101.1.247 0.0.0.0 area 0
!
ip classless
no ip http server
no ip http secure-server
!
control-plane
!
gatekeeper
 shutdown
!
line con 0
 login local
 stopbits 1
line aux 0
 stopbits 1
line vty 0 4
 exec-timeout 0 0
 password xxxx
 login
!
end
```

Router P2

```
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname P2
```

```
!
boot-start-marker
boot-end-marker
!
enable password xxxx
!
no aaa new-model
!
resource policy
!
ip subnet-zero
ip cef
!
mpls traffic-eng tunnels
mpls traffic-eng tunnels reoptimize timers frequency 60
!
no ip dhcp use vrf connected
!
no ip domain lookup
no ip ips deny-action ips-interface
!
ipv6 unicast-routing
!
username root password 0 xxxx
!
interface Loopback0
 ip address 10.101.2.247 255.255.255.255
!
interface FastEthernet0/0
 description mgm_interface
 mac-address 02fd.000c.0100
 ip address 10.250.0.206 255.255.255.252
 duplex auto
 speed auto
!
interface FastEthernet1/0
 description PE2_fa1/1
 mac-address 02fd.000c.0102
 ip address 10.1.1.21 255.255.255.252
 duplex auto
 speed auto
 mpls ip
mpls mtu 1700
 ip rsvp bandwidth 2048 2048
 mpls traffic-eng tunnels
!
interface FastEthernet1/1
 description P1_fa1/1
 mac-address 02fd.000c.0103
 ip address 10.1.1.6 255.255.255.252
 duplex auto
 speed auto
 mpls ip
mpls mtu 1700
 ip rsvp bandwidth 2048 2048
 mpls traffic-eng tunnels
```

```
!
router ospf 2914
 router-id 10.101.2.247
 log-adjacency-changes
  mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
 network 10.1.1.4 0.0.0.3 area 0
 network 10.1.1.20 0.0.0.3 area 0
 network 10.101.2.247 0.0.0.0 area 0
!
ip classless
no ip http server
no ip http secure-server
!
control-plane
!
gatekeeper
 shutdown
!
line con 0
 login local
 stopbits 1
line aux 0
 stopbits 1
line vty 0 4
 exec-timeout 0 0
 password xxxx
 login
!
end
```

Router P3

```
hostname P3
!
boot-start-marker
boot-end-marker
!
enable password xxxx
!
no aaa new-model
ip cef
!
no ip dhcp use vrf connected
!
no ip domain lookup
ipv6 unicast-routing
ipv6 cef
!
mpls traffic-eng tunnels
multilink bundle-name authenticated
!
username root password 0 xxxx
!
```

```
interface Loopback0
 ip address 10.201.1.247 255.255.255.255
!
interface FastEthernet0/0
 description mgm_interface
 mac-address 02fd.000c.0200
 ip address 10.250.0.210 255.255.255.252
 speed auto
 duplex auto
!
interface FastEthernet1/0
 description PE3_fa1/0
 mac-address 02fd.000c.0202
 ip address 10.1.1.14 255.255.255.252
 speed auto
 duplex auto
 mpls ip
 mpls mtu 1700
 mpls traffic-eng tunnels
 ip rsvp bandwidth 2048 2048
 !
interface FastEthernet1/1
 description PE2_fa1/0
 mac-address 02fd.000c.0203
 ip address 10.1.1.2 255.255.255.252
 speed auto
 duplex auto
 mpls ip
 mpls mtu 1700
 mpls traffic-eng tunnels
 ip rsvp bandwidth 2048 2048
!
router ospf 2914
 router-id 10.201.1.247
 network 10.1.1.0 0.0.0.3 area 0
 network 10.1.1.12 0.0.0.3 area 0
 network 10.201.1.247 0.0.0.0 area 0
 network 172.16.1.0 0.0.0.255 area 0
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
!
control-plane
!
line con 0
 login local
 stopbits 1
line aux 0
 stopbits 1
line vty 0 4
 exec-timeout 0 0
 password xxxx
```

```
 login
!
end
```

Router PE2

```
hostname PE2
!
boot-start-marker
boot-end-marker
!
enable password xxxx
!
no aaa new-model
ip cef
!
no ip dhcp use vrf connected
!
no ip domain lookup
ipv6 unicast-routing
ipv6 cef
!
mpls traffic-eng tunnels
multilink bundle-name authenticated
!
username root password 0 xxxx
!
interface Loopback0
 ip address 10.201.2.247 255.255.255.255
!
interface FastEthernet0/0
 description mgm_interface
 mac-address 02fd.000c.0300
 ip address 10.250.0.214 255.255.255.252
 speed auto
 duplex auto
!
interface FastEthernet0/1
 description P2_fa1/1
 mac-address 02fd.000c.0301
 ip address 10.1.1.22 255.255.255.252
 speed auto
 duplex auto
 mpls ip
 mpls mtu 1700
 mpls traffic-eng tunnels
 ip rsvp bandwidth 2048 2048
!
interface FastEthernet1/0
 description cliente_TFM
 mac-address 02fd.000c.0302
 ip address 172.16.2.1 255.255.255.0
 speed auto
 duplex auto
```

```
!
interface FastEthernet1/1
 description P3_fa1/1
 mac-address 02fd.000c.0303
 ip address 10.1.1.1 255.255.255.252
 speed auto
 duplex auto
 mpls ip
 mpls mtu 1700
 mpls traffic-eng tunnels
ip rsvp bandwidth 2048 2048
!
interface FastEthernet2/0
 description cliente_TFM
 mac-address 02fd.000c.0304
 ip address 172.16.4.1 255.255.255.0
 speed auto
 duplex auto
!
router ospf 2914
 router-id 10.201.2.247
 network 10.1.1.0 0.0.0.3 area 0
 network 10.1.1.20 0.0.0.3 area 0
 network 10.201.2.247 0.0.0.0 area 0
 network 172.16.2.0 0.0.0.255 area 0
 network 172.16.4.0 0.0.0.255 area 0
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
!
control-plane
!
line con 0
 login local
 stopbits 1
line aux 0
 stopbits 1
line vty 0 4
 exec-timeout 0 0
 password xxxx
 login
!
end
```

Router PE3

```
hostname PE3
!
boot-start-marker
boot-end-marker
```

```
!
enable password xxxx
!
no aaa new-model
ip cef
!
no ip dhcp use vrf connected
!
no ip domain lookup
ipv6 unicast-routing
ipv6 cef
!
mpls traffic-eng tunnels
mpls traffic-eng reoptimize timers frequency 60
multilink bundle-name authenticated
!
username root password 0 xxxx
!
interface Loopback0
 ip address 10.201.3.247 255.255.255.255
!
interface Tunnel2
 ip unnumbered Loopback0
 tunnel mode mpls traffic-eng
 tunnel destination 10.201.2.247
 tunnel mpls traffic-eng priority 1 1
 tunnel mpls traffic-eng bandwidth 1024
 tunnel mpls traffic-eng path-option 1 explicit name BACKUP
!
interface FastEthernet0/0
 description mgm_interface
 mac-address 02fd.000c.0400
 ip address 10.250.0.218 255.255.255.252
 speed auto
 duplex auto
!
interface FastEthernet0/1
 description cliente_TFM
 mac-address 02fd.000c.0401
 ip address 172.16.3.1 255.255.255.0
 speed auto
 duplex auto
!
interface FastEthernet1/0
 description P1_fa0/1
 mac-address 02fd.000c.0402
 ip address 10.1.1.18 255.255.255.252
 speed auto
 duplex auto
 mpls ip
 mpls mtu 1700
 mpls traffic-eng tunnels
 ip rsvp bandwidth 2048 2048
!
interface FastEthernet1/1
 description P3_fa1/0
```

```
 mac-address 02fd.000c.0403
 ip address 10.1.1.13 255.255.255.252
 speed auto
 duplex auto
 mpls ip
 mpls mtu 1700
 mpls traffic-eng tunnels
 ip rsvp bandwidth 2048 2048
!
interface FastEthernet2/0
 description cliente_TFM
 mac-address 02fd.000c.0401
 ip address 172.16.1.1 255.255.255.0
 speed auto
 duplex auto
!
router ospf 2914
 router-id 10.201.3.247
 network 10.1.1.12 0.0.0.3 area 0
 network 10.1.1.16 0.0.0.3 area 0
 network 10.201.3.247 0.0.0.0 area 0
 network 172.16.3.0 0.0.0.255 area 0
 network 172.16.1.0 0.0.0.255 area 0
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
!
ip forward-protocol nd
ip route 172.16.2.0 255.255.255.0 Tunnel2
!
no ip http server
no ip http secure-server
!
ip explicit-path name BACKUP enable
 exclude-address 10.201.1.247
!
control-plane
!
line con 0
 login local
 stopbits 1
line aux 0
 stopbits 1
line vty 0 4
 exec-timeout 0 0
 password xxxx
 login
!
!
end
```

Anexo 2. Configuraciones de los routers Experimento 2

Router P1

```
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname P1
!
boot-start-marker
boot-end-marker
!
enable password xxxx
!
no aaa new-model
!
resource policy
!
ip subnet-zero
ip cef
!
mpls traffic-eng tunnels
mpls traffic-eng tunnels timers reoptimize frequency 60
!
no ip dhcp use vrf connected
!
no ip domain lookup
no ip ips deny-action ips-interface
!
ipv6 unicast-routing
!
username root password 0 xxxx
!
interface Loopback0
ip address 10.101.1.247 255.255.255.255
!
interface FastEthernet0/0
description mgm_interface
mac-address 02fd.000c.0000
ip address 10.250.0.202 255.255.255.252
duplex auto
speed auto
!
interface FastEthernet0/1
description PE3_fa1/0
mac-address 02fd.000c.0001
ip address 10.1.1.17 255.255.255.252
duplex auto
speed auto
mpls ip
mpls mtu 1700
ip rsvp bandwidth 2048 2048
```

```
mpls traffic-eng tunnels
!
interface FastEthernet1/0
description PE1_fa1/0
ip address 10.1.1.13 255.255.255.252
shutdown
duplex auto
speed auto
mpls ip
!
interface FastEthernet1/1
description P2_fa1/1
mac-address 02fd.000c.0002
ip address 10.1.1.5 255.255.255.252
duplex auto
speed auto
mpls ip
mpls mtu 1700
ip rsvp bandwidth 2048 2048
mpls traffic-eng tunnels
!
router ospf 2914
router-id 10.101.1.247
log-adjacency-changes
 mpls traffic-eng router-id Loopback0
mpls traffic-eng area 0
network 10.1.1.4 0.0.0.3 area 0
network 10.1.1.16 0.0.0.3 area 0
network 10.101.1.247 0.0.0.0 area 0
!
ip classless
no ip http server
no ip http secure-server
!
control-plane
!
gatekeeper
shutdown
!
line con 0
login local
stopbits 1
line aux 0
stopbits 1
line vty 0 4
exec-timeout 0 0
password xxxx
login
!
end
```

Router P2

```
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname P2
!
boot-start-marker
boot-end-marker
!
enable password xxxx
!
no aaa new-model
!
resource policy
!
ip subnet-zero
ip cef
!
mpls traffic-eng tunnels
!
no ip dhcp use vrf connected
!
no ip domain lookup
no ip ips deny-action ips-interface
!
ipv6 unicast-routing
!
username root password 0 xxxx
!
interface Loopback0
 ip address 10.101.2.247 255.255.255.255
!
interface FastEthernet0/0
 description mgm_interface
 mac-address 02fd.000c.0100
 ip address 10.250.0.206 255.255.255.252
 duplex auto
 speed auto
!
interface FastEthernet1/0
 description PE2_fa1/1
 mac-address 02fd.000c.0102
 ip address 10.1.1.21 255.255.255.252
 duplex auto
 speed auto
 mpls ip
mpls mtu 1700
 ip rsvp bandwidth 2048 2048
 mpls traffic-eng tunnels
!
interface FastEthernet1/1
 description P1_fa1/1
 mac-address 02fd.000c.0103
 ip address 10.1.1.6 255.255.255.252
 duplex auto
 speed auto
```

```
 mpls ip
mpls mtu 1700
 ip rsvp bandwidth 2048 2048
 mpls traffic-eng tunnels
!
router ospf 2914
 router-id 10.101.2.247
 log-adjacency-changes
  mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
 network 10.1.1.4 0.0.0.3 area 0
 network 10.1.1.20 0.0.0.3 area 0
 network 10.101.2.247 0.0.0.0 area 0
!
ip classless
no ip http server
no ip http secure-server
!
control-plane
!
gatekeeper
 shutdown
!
line con 0
 login local
 stopbits 1
line aux 0
 stopbits 1
line vty 0 4
 exec-timeout 0 0
 password xxxx
 login
!
end
```

Router P3

```
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname P3
!
boot-start-marker
boot-end-marker
!
enable password xxxx
!
no aaa new-model
ip cef
!
no ip dhcp use vrf connected
!
no ip domain lookup
ipv6 unicast-routing
ipv6 cef
!
```

```
mpls traffic-eng tunnels
multilink bundle-name authenticated
!
username root password 0 xxxx
!
interface Loopback0
 ip address 10.201.1.247 255.255.255.255
!
interface FastEthernet0/0
 description mgm_interface
 mac-address 02fd.000c.0200
 ip address 10.250.0.210 255.255.255.252
 speed auto
 duplex auto
!
interface FastEthernet1/0
 description P1_fa1/0
 mac-address 02fd.000c.0202
 ip address 10.1.1.14 255.255.255.252
 speed auto
 duplex auto
 mpls ip
 mpls mtu 1700
 mpls traffic-eng tunnels
 ip rsvp bandwidth 2048 2048
 ip rsvp signalling hello
!
interface FastEthernet1/1
 description P2_fa1/0
 mac-address 02fd.000c.0203
 ip address 10.1.1.2 255.255.255.252
 speed auto
 duplex auto
 mpls ip
 mpls mtu 1700
 mpls traffic-eng tunnels
 ip rsvp bandwidth 2048 2048
ip rsvp signalling hello
!
router ospf 2914
 router-id 10.201.1.247
 network 10.1.1.0 0.0.0.3 area 0
 network 10.1.1.12 0.0.0.3 area 0
 network 10.201.1.247 0.0.0.0 area 0
mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
ip rsvp signalling hello
!
control-plane
!
line con 0
 login local
 stopbits 1
```

```
line aux 0
 stopbits 1
line vty 0 4
 exec-timeout 0 0
 password xxxx
 login
!
end
```

Router PE2

```
service timestamps debug datetime msec
service timestamps log datetime msec
!
hostname PE2
!
boot-start-marker
boot-end-marker
!
enable password xxxx
!
no aaa new-model
ip cef
!
no ip dhcp use vrf connected
!
no ip domain lookup
ipv6 unicast-routing
ipv6 cef
!
mpls traffic-eng tunnels
multilink bundle-name authenticated
!
username root password 0 xxxx
!
interface Loopback0
 ip address 10.201.2.247 255.255.255.255
!
nterface Tunnel1
 ip unnumbered Loopback0
 tunnel mode mpls traffic-eng
 tunnel destination 10.201.3.247
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 1 1
 tunnel mpls traffic-eng bandwidth 1024
 tunnel mpls traffic-eng path-option 1 explicit name PRINCIPAL
 tunnel mpls traffic-eng fast-reroute
!
interface Tunnel2
 ip unnumbered Loopback0
 tunnel mode mpls traffic-eng
 tunnel destination 10.201.3.247
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 1 1
 tunnel mpls traffic-eng bandwidth 1024
```

```
 tunnel mpls traffic-eng path-option 1 explicit name BACKUP
!
interface FastEthernet0/0
 description mgm_interface
 mac-address 02fd.000c.0300
 ip address 10.250.0.214 255.255.255.252
 speed auto
 duplex auto
!
interface FastEthernet0/1
 description P2_fa1/1
 mac-address 02fd.000c.0301
 ip address 10.1.1.22 255.255.255.252
 speed auto
 duplex auto
 mpls ip
 mpls mtu 1700
 mpls traffic-eng tunnels
 ip rsvp bandwidth 2048 2048
!
interface FastEthernet1/0
 description cliente_TFM
 mac-address 02fd.000c.0302
 ip address 172.16.2.1 255.255.255.0
 speed auto
 duplex auto
!
interface FastEthernet1/1
 description PE1_fa1/1
 mac-address 02fd.000c.0303
 ip address 10.1.1.1 255.255.255.252
 speed auto
 duplex auto
 mpls ip
 mpls mtu 1700
 mpls traffic-eng tunnels
 mpls traffic-eng backup-path Tunnel2
 ip rsvp bandwidth 2048 2048
 ip rsvp signalling hello
!
interface FastEthernet1/0
 description cliente_TFM
 mac-address 02fd.000c.0302
 ip address 172.16.4.1 255.255.255.0
 speed auto
 duplex auto
!
router ospf 2914
 router-id 10.201.2.247
 network 10.1.1.0 0.0.0.3 area 0
 network 10.1.1.20 0.0.0.3 area 0
 network 10.201.2.247 0.0.0.0 area 0
 network 172.16.2.0 0.0.0.255 area 0
 network 172.16.4.0 0.0.0.255 area 0
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
```

```
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
ip rsvp signalling hello
!
ip explicit-path name PRINCIPAL enable
 next-address 10.1.1.2
 next-address 10.1.1.13
!
ip explicit-path name BACKUP enable
 exclude-address 10.201.1.247
!
control-plane
!
line con 0
 login local
 stopbits 1
line aux 0
 stopbits 1
line vty 0 4
 exec-timeout 0 0
 password xxxx
 login
!
!
end
```

Router PE3

```
service timestamps debug datetime msec
service timestamps log datetime msec
!
hostname PE3
!
boot-start-marker
boot-end-marker
!
enable password xxxx
!
no aaa new-model
ip cef
!
no ip dhcp use vrf connected
!
no ip domain lookup
ipv6 unicast-routing
ipv6 cef
!
mpls traffic-eng tunnels
multilink bundle-name authenticated
!
username root password 0 xxxx
!
```

```
interface Loopback0
 ip address 10.201.3.247 255.255.255.255
!
interface Tunnel1
 ip unnumbered Loopback0
 tunnel mode mpls traffic-eng
 tunnel destination 10.201.2.247
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 1 1
 tunnel mpls traffic-eng bandwidth 1024
 tunnel mpls traffic-eng path-option 1 explicit name PRINCIPAL
 tunnel mpls traffic-eng fast-reroute
!
interface Tunnel2
 ip unnumbered Loopback0
 tunnel mode mpls traffic-eng
 tunnel destination 10.201.2.247
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 1 1
 tunnel mpls traffic-eng bandwidth 1024
 tunnel mpls traffic-eng path-option 1 explicit name BACKUP
!
interface FastEthernet0/0
 description mgm_interface
 mac-address 02fd.000c.0400
 ip address 10.250.0.218 255.255.255.252
 speed auto
 duplex auto
!
interface FastEthernet0/1
 description cliente_TFM
 mac-address 02fd.000c.0401
 ip address 172.16.3.1 255.255.255.0
 speed auto
 duplex auto
!
interface FastEthernet1/0
 description P1_fa0/1
 mac-address 02fd.000c.0402
 ip address 10.1.1.18 255.255.255.252
 speed auto
 duplex auto
 mpls ip
 mpls mtu 1700
 mpls traffic-eng tunnels
 ip rsvp bandwidth 2048 2048
!
interface FastEthernet1/1
 description PE1_fa1/0
 mac-address 02fd.000c.0403
 ip address 10.1.1.13 255.255.255.252
 speed auto
 duplex auto
 mpls ip
 mpls mtu 1700
 mpls traffic-eng tunnels
```

```
 mpls traffic-eng backup-path Tunnel2
 ip rsvp bandwidth 2048 2048
 ip rsvp signalling hello
!
interface FastEthernet2/0
 mac-address 02fd.000c.0404
 ip address 172.16.1.1 255.255.255.0
 speed auto
 duplex auto
!
interface FastEthernet2/1
 no ip address
 shutdown
 speed auto
 duplex auto
!
router ospf 2914
 router-id 10.201.3.247
 network 10.1.1.12 0.0.0.3 area 0
 network 10.1.1.16 0.0.0.3 area 0
 network 10.201.3.247 0.0.0.0 area 0
 network 172.16.3.0 0.0.0.255 area 0
 network 172.16.1.0 0.0.0.255 area 0
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
!
ip forward-protocol nd
!
!
no ip http server
no ip http secure-server
ip rsvp signalling hello
!
ip explicit-path name PRINCIPAL enable
 next-address 10.1.1.14
 next-address 10.1.1.1
!
ip explicit-path name BACKUP enable
 exclude-address 10.201.1.247
!
control-plane
!
line con 0
 login local
 stopbits 1
line aux 0
 stopbits 1
line vty 0 4
 exec-timeout 0 0
 password xxxx
 login
!
end
```

Anexo 3. Configuraciones de los routers Experimento 3

Router P1

```
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname P1
!
boot-start-marker
boot-end-marker
!
enable password xxxx
!
no aaa new-model
!
resource policy
!
ip subnet-zero
ip cef
!
mpls traffic-eng tunnels
mpls traffic-eng tunnels reoptimize timers frequency 60
!
no ip dhcp use vrf connected
!
no ip domain lookup
no ip ips deny-action ips-interface
!
ipv6 unicast-routing
!
username root password 0 xxxx
!
interface Loopback0
 ip address 10.101.1.247 255.255.255.255
!
interface FastEthernet0/0
 description mgm_interface
 mac-address 02fd.000c.0000
 ip address 10.250.0.202 255.255.255.252
 duplex auto
 speed auto
!
interface FastEthernet0/1
 description PE3_fa1/0
 mac-address 02fd.000c.0001
 ip address 10.1.1.17 255.255.255.252
 duplex auto
 speed auto
 mpls ip
 mpls mtu 1700
 ip rsvp bandwidth 2048 2048
```

```
 mpls traffic-eng tunnels
!
interface FastEthernet1/0
 description PE1_fa1/0
 ip address 10.1.1.13 255.255.255.252
 shutdown
 duplex auto
 speed auto
 mpls ip
!
interface FastEthernet1/1
 description P2_fa1/1
 mac-address 02fd.000c.0002
 ip address 10.1.1.5 255.255.255.252
 duplex auto
 speed auto
 mpls ip
mpls mtu 1700
ip rsvp bandwidth 2048 2048
mpls traffic-eng tunnels
!
router ospf 2914
 router-id 10.101.1.247
 log-adjacency-changes
  mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
 network 10.1.1.4 0.0.0.3 area 0
 network 10.1.1.16 0.0.0.3 area 0
 network 10.101.1.247 0.0.0.0 area 0
!
ip classless
no ip http server
no ip http secure-server
!
!
control-plane
!
!
gatekeeper
 shutdown
!
!
line con 0
 login local
 stopbits 1
line aux 0
 stopbits 1
line vty 0 4
 exec-timeout 0 0
 password xxxx
 login
!
!
end
```

Router P2

```
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname P2
!
boot-start-marker
boot-end-marker
!
enable password xxxx
!
no aaa new-model
!
resource policy
!
ip subnet-zero
ip cef
!
mpls traffic-eng tunnels
mpls traffic-eng tunnels reoptimize timers frequency 60
!
no ip dhcp use vrf connected
!
no ip domain lookup
no ip ips deny-action ips-interface
!
ipv6 unicast-routing
!
username root password 0 xxxx
!
interface Loopback0
 ip address 10.101.2.247 255.255.255.255
!
interface FastEthernet0/0
 description mgm_interface
 mac-address 02fd.000c.0100
 ip address 10.250.0.206 255.255.255.252
 duplex auto
 speed auto
!
interface FastEthernet1/0
 description PE2_fa1/1
 mac-address 02fd.000c.0102
 ip address 10.1.1.21 255.255.255.252
 duplex auto
 speed auto
 mpls ip
mpls mtu 1700
 ip rsvp bandwidth 2048 2048
 mpls traffic-eng tunnels
!
interface FastEthernet1/1
 description P1_fa1/1
```

```
 mac-address 02fd.000c.0103
 ip address 10.1.1.6 255.255.255.252
 duplex auto
 speed auto
 mpls ip
mpls mtu 1700
 ip rsvp bandwidth 2048 2048
 mpls traffic-eng tunnels
!
router ospf 2914
 router-id 10.101.2.247
 log-adjacency-changes
  mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
 network 10.1.1.4 0.0.0.3 area 0
 network 10.1.1.20 0.0.0.3 area 0
 network 10.101.2.247 0.0.0.0 area 0
!
ip classless
no ip http server
no ip http secure-server
!
control-plane
!
gatekeeper
 shutdown
!
line con 0
 login local
 stopbits 1
line aux 0
 stopbits 1
line vty 0 4
 exec-timeout 0 0
 password xxxx
 login
!
end
```

Router P3

```
hostname P3
!
boot-start-marker
boot-end-marker
!
enable password xxxx
!
no aaa new-model
ip cef
!
no ip dhcp use vrf connected
!
no ip domain lookup
```

```
ipv6 unicast-routing
ipv6 cef
!
mpls traffic-eng tunnels
multilink bundle-name authenticated
!
username root password 0 xxxx
!
interface Loopback0
 ip address 10.201.1.247 255.255.255.255
!
interface FastEthernet0/0
 description mgm_interface
 mac-address 02fd.000c.0200
 ip address 10.250.0.210 255.255.255.252
 speed auto
 duplex auto
!
interface FastEthernet1/0
 description P1_fa1/0
 mac-address 02fd.000c.0202
 ip address 10.1.1.14 255.255.255.252
 speed auto
 duplex auto
 mpls ip
 mpls mtu 1700
 mpls traffic-eng tunnels
 ip rsvp bandwidth 2048 2048
!
interface FastEthernet1/1
 description P2_fa1/0
 mac-address 02fd.000c.0203
 ip address 10.1.1.2 255.255.255.252
 speed auto
 duplex auto
 mpls ip
 mpls mtu 1700
 mpls traffic-eng tunnels
 ip rsvp bandwidth 2048 2048
!
router ospf 2914
 router-id 10.201.1.247
 network 10.1.1.0 0.0.0.3 area 0
 network 10.1.1.12 0.0.0.3 area 0
 network 10.201.1.247 0.0.0.0 area 0
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
!
control-plane
!
line con 0
```

```
 login local
 stopbits 1
line aux 0
 stopbits 1
line vty 0 4
 exec-timeout 0 0
 password xxxx
 login
!
!
end
```

Router PE2

```
hostname PE2
!
boot-start-marker
boot-end-marker
!
enable password xxxx
!
no aaa new-model
ip cef
!
no ip dhcp use vrf connected
!
no ip domain lookup
ipv6 unicast-routing
ipv6 cef
!
mpls traffic-eng tunnels
multilink bundle-name authenticated
!
username root password 0 xxxx
!
interface Loopback0
 ip address 10.201.2.247 255.255.255.255
!
interface FastEthernet0/0
 description mgm_interface
 mac-address 02fd.000c.0300
 ip address 10.250.0.214 255.255.255.252
 speed auto
 duplex auto
!
interface FastEthernet0/1
 description P2_fa1/1
 mac-address 02fd.000c.0301
 ip address 10.1.1.22 255.255.255.252
 speed auto
 duplex auto
 mpls ip
 mpls mtu 1700
 mpls traffic-eng tunnels
```

```
 ip rsvp bandwidth 2048 2048
!
interface FastEthernet1/0
 description cliente_TFM
 mac-address 02fd.000c.0302
 ip address 172.16.2.1 255.255.255.0
 speed auto
 duplex auto
!
interface FastEthernet1/1
 description PE1_fa1/1
 mac-address 02fd.000c.0303
 ip address 10.1.1.1 255.255.255.252
 speed auto
 duplex auto
 mpls ip
 mpls mtu 1700
 mpls traffic-eng tunnels
ip rsvp bandwidth 2048 2048
!
interface FastEthernet2/0
 description cliente_TFM
 mac-address 02fd.000c.0304
 ip address 172.16.4.1 255.255.255.0
 speed auto
 duplex auto
!
router ospf 2914
 router-id 10.201.2.247
 network 10.1.1.0 0.0.0.3 area 0
 network 10.1.1.20 0.0.0.3 area 0
 network 10.201.2.247 0.0.0.0 area 0
 network 172.16.2.0 0.0.0.255 area 0
 network 172.16.4.0 0.0.0.255 area 0
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
!
control-plane
!
line con 0
 login local
 stopbits 1
line aux 0
 stopbits 1
line vty 0 4
 exec-timeout 0 0
 password xxxx
 login
!
end
```

Router PE3

```
hostname PE3
!
boot-start-marker
boot-end-marker
!
enable password xxxx
!
no aaa new-model
ip cef
!
no ip dhcp use vrf connected
!
no ip domain lookup
ipv6 unicast-routing
ipv6 cef
!
mpls traffic-eng tunnels
mpls traffic-eng reoptimize timers frequency 60
multilink bundle-name authenticated
!
username root password 0 xxxx
!
interface Loopback0
 ip address 10.201.3.247 255.255.255.255
!
interface Tunnel1
 ip unnumbered Loopback0
 tunnel mode mpls traffic-eng
 tunnel destination 10.201.2.247
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 1 1
 tunnel mpls traffic-eng bandwidth 512
 tunnel mpls traffic-eng path-option 1 explicit name PRINCIPAL
 !
interface Tunnel2
 ip unnumbered Loopback0
 tunnel mode mpls traffic-eng
 tunnel destination 10.201.2.247
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng priority 5 5
 tunnel mpls traffic-eng bandwidth 1024
 tunnel mpls traffic-eng path-option 1 explicit name BACKUP
 !
interface FastEthernet0/0
 description mgm_interface
 mac-address 02fd.000c.0400
 ip address 10.250.0.218 255.255.255.252
 speed auto
 duplex auto
!
interface FastEthernet0/1
 description cliente_TFM
 mac-address 02fd.000c.0401
```

```
 ip address 172.16.3.1 255.255.255.0
 speed auto
 duplex auto
!
interface FastEthernet1/0
 description P1_fa0/1
 mac-address 02fd.000c.0402
 ip address 10.1.1.18 255.255.255.252
 speed auto
 duplex auto
 mpls ip
 mpls mtu 1700
 mpls traffic-eng tunnels
 ip rsvp bandwidth 2048 2048
!
interface FastEthernet1/1
 description PE1_fa1/0
 mac-address 02fd.000c.0403
 ip address 10.1.1.13 255.255.255.252
 speed auto
 duplex auto
 mpls ip
 mpls mtu 1700
 mpls traffic-eng tunnels
 ip rsvp bandwidth 2048 2048
!
interface FastEthernet2/0
 description cliente_TFM
 mac-address 02fd.000c.0404
 ip address 172.16.1.1 255.255.255.0
 speed auto
 duplex auto

!
router ospf 2914
 router-id 10.201.3.247
 network 10.1.1.12 0.0.0.3 area 0
 network 10.1.1.16 0.0.0.3 area 0
 network 10.201.3.247 0.0.0.0 area 0
 network 172.16.3.0 0.0.0.255 area 0
 network 172.16.1.0 0.0.0.255 area 0
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
!
ip explicit-path name PRINCIPAL enable
 next-address 10.1.1.14
 next-address 10.1.1.1
!
ip explicit-path name BACKUP enable
 exclude-address 10.201.1.247
!
```

```
control-plane
!
line con 0
 login local
 stopbits 1
line aux 0
 stopbits 1
line vty 0 4
 exec-timeout 0 0
 password xxxx
 login
!
!
end
```

Anexo 4. Configuraciones de los routers Experimento 4

Router P1

```
service timestamps debug datetime msec
service timestamps log datetime msec
!
hostname P1
!
boot-start-marker
boot-end-marker
!
enable password xxxx
!
no aaa new-model
ip cef
!
no ip dhcp use vrf connected
!
no ip domain lookup
ipv6 unicast-routing
ipv6 cef
!
mpls traffic-eng tunnels
mpls traffic-eng reoptimize timers frequency 60
multilink bundle-name authenticated
!
username root password 0 xxxx
!
interface Loopback0
 ip address 10.101.1.247 255.255.255.255
!
interface FastEthernet0/0
 description mgm_interface
 mac-address 02fd.000c.0000
 ip address 10.250.0.202 255.255.255.252
 speed auto
 duplex auto
!
interface FastEthernet0/1
 description PE3_fa1/0
 mac-address 02fd.000c.0001
 ip address 10.1.1.17 255.255.255.252
 speed auto
 duplex auto
 mpls ip
 mpls mtu 1700
 mpls traffic-eng tunnels
 ip rsvp bandwidth 3072 sub-pool 2048
!
interface FastEthernet1/0
 description PE1_fa1/0
 ip address 10.1.1.13 255.255.255.252
```

```
 shutdown
 speed auto
 duplex auto
 mpls ip
!
interface FastEthernet1/1
 description P2_fa1/1
 mac-address 02fd.000c.0002
 ip address 10.1.1.5 255.255.255.252
 speed auto
 duplex auto
 mpls ip
 mpls mtu 1700
 mpls traffic-eng tunnels
 ip rsvp bandwidth 3072 sub-pool 2048
!
router ospf 2914
 router-id 10.101.1.247
 network 10.1.1.4 0.0.0.3 area 0
 network 10.1.1.16 0.0.0.3 area 0
 network 10.101.1.247 0.0.0.0 area 0
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
!
control-plane
!
line con 0
 login local
 stopbits 1
line aux 0
 stopbits 1
line vty 0 4
 exec-timeout 0 0
 password xxxx
 login
!
end
```

Router P2

```
hostname P2
!
boot-start-marker
boot-end-marker
!
enable password xxxx
!
no aaa new-model
ip cef
```

```
!
no ip dhcp use vrf connected
!
no ip domain lookup
ipv6 unicast-routing
ipv6 cef
!
mpls traffic-eng tunnels
mpls traffic-eng reoptimize timers frequency 60
multilink bundle-name authenticated
!
username root password 0 xxxx
!
interface Loopback0
 ip address 10.101.2.247 255.255.255.255
!
interface FastEthernet0/0
 description mgm_interface
 mac-address 02fd.000c.0100
 ip address 10.250.0.206 255.255.255.252
 speed auto
 duplex auto
!
interface FastEthernet0/1
 mac-address 02fd.000c.0101
 ip address 10.1.1.10 255.255.255.252
 speed auto
 duplex auto
!
interface FastEthernet1/0
 description PE2_fa1/1
 mac-address 02fd.000c.0102
 ip address 10.1.1.21 255.255.255.252
 speed auto
 duplex auto
 mpls ip
 mpls mtu 1700
 mpls traffic-eng tunnels
 ip rsvp bandwidth 3072 sub-pool 2048
!
interface FastEthernet1/1
 description P1_fa1/1
 mac-address 02fd.000c.0103
 ip address 10.1.1.6 255.255.255.252
 speed auto
 duplex auto
 mpls ip
 mpls mtu 1700
 mpls traffic-eng tunnels
 ip rsvp bandwidth 3072 sub-pool 2048
!
router ospf 2914
 router-id 10.101.2.247
 network 10.1.1.4 0.0.0.3 area 0
 network 10.1.1.20 0.0.0.3 area 0
 network 10.101.2.247 0.0.0.0 area 0
```

```
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
!
control-plane
!
line con 0
 login local
 stopbits 1
line aux 0
 stopbits 1
line vty 0 4
 exec-timeout 0 0
 password xxxx
 login
!
end
```

Router P3

```
hostname P3
!
boot-start-marker
boot-end-marker
!
enable password xxxx
!
no aaa new-model
ip cef
!
no ip dhcp use vrf connected
!
no ip domain lookup
ipv6 unicast-routing
ipv6 cef
!
mpls traffic-eng tunnels
mpls traffic-eng reoptimize timers frequency 60
multilink bundle-name authenticated
!
username root password 0 xxxx
!
interface Loopback0
 ip address 10.201.1.247 255.255.255.255
!
interface FastEthernet0/0
 description mgm_interface
 mac-address 02fd.000c.0200
 ip address 10.250.0.210 255.255.255.252
 speed auto
```

```
 duplex auto
!
interface FastEthernet0/1
 mac-address 02fd.000c.0201
 no ip address
 shutdown
 speed auto
 duplex auto
!
interface FastEthernet1/0
 description P1_fa1/0
 mac-address 02fd.000c.0202
 ip address 10.1.1.14 255.255.255.252
 speed auto
 duplex auto
 mpls ip
 mpls mtu 1700
 mpls traffic-eng tunnels
 ip rsvp bandwidth 3072 sub-pool 2048
!
interface FastEthernet1/1
 description P2_fa1/0
 mac-address 02fd.000c.0203
 ip address 10.1.1.2 255.255.255.252
 speed auto
 duplex auto
 mpls ip
 mpls mtu 1700
 mpls traffic-eng tunnels
 ip rsvp bandwidth 3072 sub-pool 2048
!
router ospf 2914
 router-id 10.201.1.247
 network 10.1.1.0 0.0.0.3 area 0
 network 10.1.1.12 0.0.0.3 area 0
 network 10.201.1.247 0.0.0.0 area 0
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
!
control-plane
!
line con 0
 login local
 stopbits 1
line aux 0
 stopbits 1
line vty 0 4
 exec-timeout 0 0
 password xxxx
 login
!
```

```
end
```
Router PE2

```
hostname PE2
!
boot-start-marker
boot-end-marker
!
enable password xxxx
!
no aaa new-model
ip cef
!
no ip dhcp use vrf connected
!
no ip domain lookup
ipv6 unicast-routing
ipv6 cef
!
mpls traffic-eng tunnels
mpls traffic-eng reoptimize timers frequency 60
multilink bundle-name authenticated
!
username root password 0 xxxx
!
interface Loopback0
 ip address 10.201.2.247 255.255.255.255
!
interface FastEthernet0/0
 description mgm_interface
 mac-address 02fd.000c.0300
 ip address 10.250.0.214 255.255.255.252
 speed auto
 duplex auto
!
interface FastEthernet0/1
 description P2_fa1/1
 mac-address 02fd.000c.0301
 ip address 10.1.1.22 255.255.255.252
 speed auto
 duplex auto
 mpls ip
 mpls mtu 1700
 mpls traffic-eng tunnels
 ip rsvp bandwidth 3072 sub-pool 2048
!
interface FastEthernet1/0
 description cliente_TFM
 mac-address 02fd.000c.0302
 ip address 172.16.2.1 255.255.255.0
 speed auto
 duplex auto
!
interface FastEthernet1/1
 description PE1_fa1/1
```

```
 mac-address 02fd.000c.0303
 ip address 10.1.1.1 255.255.255.252
 speed auto
 duplex auto
 mpls ip
 mpls mtu 1700
 mpls traffic-eng tunnels
 ip rsvp bandwidth 3072 sub-pool 2048
!
interface FastEthernet2/0
 description cliente_TFM
 mac-address 02fd.000c.0304
 ip address 172.16.4.1 255.255.255.0
 speed auto
 duplex auto
!
router ospf 2914
 router-id 10.201.2.247
 network 10.1.1.0 0.0.0.3 area 0
 network 10.1.1.20 0.0.0.3 area 0
 network 10.201.2.247 0.0.0.0 area 0
 network 172.16.2.0 0.0.0.255 area 0
 network 172.16.4.0 0.0.0.255 area 0
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
!
control-plane
!
line con 0
 login local
 stopbits 1
line aux 0
 stopbits 1
line vty 0 4
 exec-timeout 0 0
 password xxxx
 login
!
end
```

Router PE3

```
hostname PE3
!
boot-start-marker
boot-end-marker
!
enable password xxxx
!
```

```
no aaa new-model
ip cef
!
no ip dhcp use vrf connected
!
no ip domain lookup
ipv6 unicast-routing
ipv6 cef
!
mpls traffic-eng tunnels
mpls traffic-eng reoptimize timers frequency 60
multilink bundle-name authenticated
!
username root password 0 xxxx
!
class-map match-all exp-5-voz
 match mpls experimental topmost 5
class-map match-all voz
 match access-group 100
class-map match-all datos
 match access-group 101
!
policy-map voz-y-datos
 class exp-5-voz
  priority percent 40
 class class-default
  bandwidth percent 60
policy-map voz
 class voz
  police cir 1229000 bc 122900 be 122900
   conform-action set-mpls-exp-imposition-transmit 5
   exceed-action drop
 class class-default
  set mpls experimental imposition 0
!
interface Loopback0
 ip address 10.201.3.247 255.255.255.255
!
interface Tunnel1
 ip unnumbered Loopback0
 tunnel mode mpls traffic-eng
 tunnel destination 10.201.2.247
 tunnel mpls traffic-eng priority 0 0
 tunnel mpls traffic-eng bandwidth sub-pool 1229
 tunnel mpls traffic-eng path-option 1 explicit name PRINCIPAL
!
interface FastEthernet0/0
 description mgm_interface
 mac-address 02fd.000c.0400
 ip address 10.250.0.218 255.255.255.252
 speed auto
 duplex auto
!
interface FastEthernet0/1
 description cliente_H3
 mac-address 02fd.000c.0401
```

```
 ip address 172.16.3.1 255.255.255.0
 speed auto
 duplex auto
 service-policy input voz
!
interface FastEthernet1/0
 description P1_fa0/1
 mac-address 02fd.000c.0402
 ip address 10.1.1.18 255.255.255.252
 load-interval 30
 speed auto
 duplex auto
 mpls ip
 mpls mtu 1700
 mpls traffic-eng tunnels
 service-policy output voz-y-datos
 ip rsvp bandwidth 3072 sub-pool 2048
!
interface FastEthernet1/1
 description PE1_fa1/0
 mac-address 02fd.000c.0403
 ip address 10.1.1.13 255.255.255.252
 load-interval 30
 speed auto
 duplex auto
 mpls ip
 mpls mtu 1700
 mpls traffic-eng tunnels
 service-policy output voz-y-datos
 ip rsvp bandwidth 3072 sub-pool 2048
!
interface FastEthernet2/0
 description cliente_H1
 mac-address 02fd.000c.0404
 ip address 172.16.1.1 255.255.255.0
 speed auto
 duplex auto
 service-policy input voz
!
interface FastEthernet2/1
 no ip address
 shutdown
 speed auto
 duplex auto
!
router ospf 2914
 router-id 10.201.3.247
 network 10.1.1.12 0.0.0.3 area 0
 network 10.1.1.16 0.0.0.3 area 0
 network 10.201.3.247 0.0.0.0 area 0
 network 172.16.1.0 0.0.0.255 area 0
 network 172.16.3.0 0.0.0.255 area 0
 mpls traffic-eng router-id Loopback0
 mpls traffic-eng area 0
!
ip forward-protocol nd
```

```
!
no ip http server
no ip http secure-server
ip route 172.16.2.2 255.255.255.255 Tunnel1
!
ip explicit-path name PRINCIPAL enable
 exclude-address 10.101.1.247
!
access-list 100 permit ip any host 172.16.2.2
access-list 101 deny   ip any host 172.16.2.2
access-list 101 permit ip any any
!
control-plane
!
line con 0
 login local
 stopbits 1
line aux 0
 stopbits 1
line vty 0 4
 exec-timeout 0 0
 password xxxx
 login
!
end
```

Anexo 5. Configuración de escenario en VNX

```
<?xml version="1.0" encoding="UTF-8"?>

<!--

~~~~~~~~~~~~~~~~~~~~
VNX Sample scenarios
~~~~~~~~~~~~~~~~~~~~

Name:        tutorial_ubuntu
Description: As simple tutorial scenario made of 6 Ubuntu
virtual machines (4 hosts: h1, h2, h3 and h4;
             and 2 routers: r1 and r2) connected through three
virtual networks. The host participates
             in the scenario having a network interface in Net3.

This file is part of the Virtual Networks over LinuX (VNX)
Project distribution.
(www: http://www.dit.upm.es/vnx - e-mail: vnx@dit.upm.es)

Departamento de Ingenieria de Sistemas Telematicos (DIT)
Universidad Politecnica de Madrid
SPAIN

-->

<vnx xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="/usr/share/xml/vnx/vnx-
2.00.xsd">
  <global>
    <version>2.0</version>
    <scenario_name>tutorial_mpls_te_labfinal</scenario_name>
    <automac offset="12"/>
   <!-- <vm_mgmt type="none" />-->
    <vm_mgmt type="private" network="10.250.0.0" mask="24"
offset="200">
       <host_mapping />
    </vm_mgmt>
    <vm_defaults>
        <console id="0" display="no"/>
        <console id="1" display="yes"/>
    </vm_defaults>
     <dynamips_ext>tutorial_mpls_te_labfinalte-qos-
dn.xml</dynamips_ext>
  </global>

  <net name="Net0" mode="virtual_bridge" />
  <net name="Net1" mode="virtual_bridge" />
  <net name="Net2" mode="virtual_bridge" />
  <net name="Net3" mode="virtual_bridge" />
  <net name="Net4" mode="virtual_bridge" />
  <net name="Net5" mode="virtual_bridge" />
```

```
  <net name="Net6" mode="virtual_bridge" />
  <net name="Net7" mode="virtual_bridge" />
  <net name="Net8" mode="virtual_bridge" />
  <net name="br0" mode="virtual_bridge" managed="no"/>

  <vm name="P1" type="dynamips" subtype="7200" os="">
    <filesystem
type="cow">/usr/share/vnx/filesystems/c7200FRR</filesystem>
      <mem>256M</mem>
      <console id="1" display="yes"/>
      <if id="0" net="vm_mgmt" name="fa0/0"/>
      <if id="1" net="Net2" name="fa0/1">
           <ipv4>10.1.1.17/30</ipv4>
      </if>
      <if id="2" net="Net4" name="fa1/1">
           <ipv4>10.1.1.5/30</ipv4>
      </if>
  </vm>

<vm name="P2" type="dynamips" subtype="7200" os="">
    <filesystem
type="cow">/usr/share/vnx/filesystems/c7200FRR</filesystem>
      <mem>256M</mem>
      <console id="1" display="yes"/>
      <if id="0" net="vm_mgmt" name="fa0/0"/>
      <if id="1" net="Net7" name="fa1/0">
           <ipv4>10.1.1.21/30</ipv4>
      </if>
         <if id="2" net="Net4" name="fa1/1">
                 <ipv4>10.1.1.6/30</ipv4>
         </if>

</vm>

<vm name="P3" type="dynamips" subtype="7200" os="">
    <filesystem
type="cow">/usr/share/vnx/filesystems/c7200FRR</filesystem>
      <mem>256M</mem>
      <console id="1" display="yes"/>
      <if id="0" net="vm_mgmt" name="fa0/0"/>
      <if id="2" net="Net3" name="fa1/0">
           <ipv4>10.1.1.14/30</ipv4>
      </if>
      <if id="3" net="Net6" name="fa1/1">
           <ipv4>10.1.1.2/30</ipv4>
      </if>
</vm>

<vm name="PE2" type="dynamips" subtype="7200" os="">
    <filesystem
type="cow">/usr/share/vnx/filesystems/c7200FRR</filesystem>
      <mem>256M</mem>
      <console id="1" display="yes"/>
      <if id="0" net="vm_mgmt" name="fa0/0"/>
      <if id="1" net="Net7" name="fa0/1">
           <ipv4>10.1.1.22/30</ipv4>
```

```
      </if>
       <if id="2" net="Net8" name="fa1/0">
                <ipv4>172.16.2.1/24</ipv4>
         </if>
         <if id="3" net="Net6" name="fa1/1">
                <ipv4>10.1.1.1/30</ipv4>
         </if>
      <if id="4" net="Net5" name="fa2/0">
                <ipv4>172.16.4.1/24</ipv4>
         </if>

</vm>

<vm name="PE3" type="dynamips" subtype="7200" os="">
    <filesystem
type="cow">/usr/share/vnx/filesystems/c7200FRR</filesystem>
      <mem>256M</mem>
      <console id="1" display="yes"/>
      <if id="0" net="vm_mgmt" name="fa0/0"/>
      <if id="1" net="Net1" name="fa0/1">
           <ipv4>172.16.3.1/24</ipv4>
      </if>
      <if id="2" net="Net2" name="fa1/0">
           <ipv4>10.1.1.18/30</ipv4>
      </if>
         <if id="3" net="Net3" name="fa1/1">
                <ipv4>10.1.1.13/30</ipv4>
         </if>
         <if id="4" net="Net0" name="fa2/0">
                <ipv4>172.16.1.1/24</ipv4>
         </if>


</vm>


  <vm name="H1" type="lxc">
   <filesystem
type="cow">/usr/share/vnx/filesystems/rootfs_lxc</filesystem>
   <if id="1" net="Net0">
      <ipv4>172.16.1.2/24</ipv4>
    </if>
    <if id="2" net="br0">
        <ipv4>192.168.37.10/24</ipv4>
    </if>
    <route type="ipv4" gw="172.16.1.1">default</route>
    <!-- Copy the files under conf/tutorial_ubuntu/h3 to vm
/var/www directory -->
    <filetree seq="start-www"
root="/var/www/">conf/tutorial_ubuntu/h3</filetree>
    <!-- Start/stop apache www server -->
    <exec seq="start-www" type="verbatim" ostype="system">chmod
644 /var/www/*</exec>
    <exec seq="start-www" type="verbatim"
ostype="system">service apache2 start</exec>
```

```
    <exec seq="stop-www" type="verbatim"
ostype="system">service apache2 stop</exec>
  </vm>

 <vm name="H2" type="lxc">
   <filesystem
type="cow">/usr/share/vnx/filesystems/rootfs_lxc</filesystem>
   <if id="1" net="Net8">
      <ipv4>172.16.2.2/24</ipv4>
    </if>
    <if id="2" net="br0">
        <ipv4>192.168.37.20/24</ipv4>
    </if>
    <route type="ipv4" gw="172.16.2.1">default</route>
    <!-- Copy the files under conf/tutorial_ubuntu/h3 to vm
/var/www directory -->
    <filetree seq="start-www"
root="/var/www/">conf/tutorial_ubuntu/h3</filetree>
    <!-- Start/stop apache www server -->
    <exec seq="start-www" type="verbatim" ostype="system">chmod
644 /var/www/*</exec>
    <exec seq="start-www" type="verbatim"
ostype="system">service apache2 start</exec>
    <exec seq="stop-www" type="verbatim"
ostype="system">service apache2 stop</exec>
  </vm>


  <vm name="H3" type="lxc">
    <filesystem
type="cow">/usr/share/vnx/filesystems/rootfs_lxc</filesystem>
    <if id="1" net="Net1">
      <ipv4>172.16.3.2/24</ipv4>
    </if>
    <if id="2" net="br0">
        <ipv4>192.168.37.30/24</ipv4>
    </if>
    <route type="ipv4" gw="172.16.3.1">default</route>
    <!-- Copy the files under conf/tutorial_ubuntu/h4 to vm
/var/www directory -->
    <filetree seq="start-www"
root="/var/www/">conf/tutorial_ubuntu/h4</filetree>
    <!-- Start/stop apache www server -->
    <exec seq="start-www" type="verbatim" ostype="system">chmod
644 /var/www/*</exec>
    <exec seq="start-www" type="verbatim"
ostype="system">service apache2 start</exec>
    <exec seq="stop-www" type="verbatim"
ostype="system">service apache2 stop</exec>
  </vm>

<vm name="H4" type="lxc">
    <filesystem
type="cow">/usr/share/vnx/filesystems/rootfs_lxc</filesystem>
    <if id="1" net="Net5">
      <ipv4>172.16.4.2/24</ipv4>
```

```
    </if>
    <route type="ipv4" gw="172.16.4.1">default</route>
    <!-- Copy the files under conf/tutorial_ubuntu/h4 to vm
/var/www directory -->
    <filetree seq="start-www"
root="/var/www/">conf/tutorial_ubuntu/h4</filetree>
    <!-- Start/stop apache www server -->
    <exec seq="start-www" type="verbatim" ostype="system">chmod
644 /var/www/*</exec>
    <exec seq="start-www" type="verbatim"
ostype="system">service apache2 start</exec>
    <exec seq="stop-www" type="verbatim"
ostype="system">service apache2 stop</exec>
  </vm>


</vnx>
```

Anexo 6. Código del programa "cambia_mtu.bin"

```
#!/bin/bash

sudo ip link set Net2-e00 mtu 1700
sudo ip link set P1-e1 mtu 1700
sudo ip link set PE3-e2 mtu 1700

sudo ip link set Net3-e00 mtu 1700
sudo ip link set PE1-e2 mtu 1700
sudo ip link set PE3-e3 mtu 1700

sudo ip link set Net4-e00 mtu 1700
sudo ip link set P1-e2 mtu 1700
sudo ip link set P2-e2 mtu 1700

sudo ip link set Net6-e00 mtu 1700
sudo ip link set PE1-e3 mtu 1700
sudo ip link set PE2-e3 mtu 1700

sudo ip link set Net7-e00 mtu 1700
sudo ip link set P2-e2 mtu 1700
sudo ip link set PE2-e1 mtu 1700

sudo ip link set Net7-e00 mtu 1700
sudo ip link set P2-e2 mtu 1700
sudo ip link set PE2-e1 mtu 1700
```