PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR

FACULTAD DE INGENIERÍA

TESIS PARA OBTENCIÓN DEL TÍTULO DE

MAGISTER EN TECNOLOGÍAS DE LA INFORMACIÓN
MENCIÓN REDES DE COMUNICACIONES

TEMA:

"EMULACIÓN Y EVALUACIÓN DE SEGMENT ROUTING IPV6 PARA SU FACTIBILIDAD DE IMPLEMENTACIÓN EN SERVICE PROVIDERS"

AUTOR:

ING. ANDY RAFAEL REINOSO GARCÍA

DIRECTOR:

ING. GUSTAVO SALAZAR CHACÓN, MSc.

Quito – 2021

AUTORÍA

Yo, Andy Rafael Reinoso García portador de la cédula de ciudadanía No.1714311998, declaro bajo juramento que la presente investigación es de total responsabilidad del autor, y que se ha respetado las diferentes fuentes de información realizando las citas correspondientes. Esta investigación no contiene plagio alguno y es resultado de un trabajo serio desarrollado en su totalidad por mi persona.

Firmado electrónicamente por:
ANDY RAFAEL REINOSO GARCIA

Andy Rafael Reinoso García

DEDICATORIA

Para mi angelita bella

Priscila Charlotte,

que siempre me acompaña,

y alegra mi alma

AGRADECIMIENTO

A Dios, por brindarme salud y vida, a mis padres Víctor y Priscila en quienes siempre confío y que con su amor me apoyan en los proyectos que emprendo, a mis hermanos Cristhian y Lourdes y mis sobrinos Toñito, Amelia, Antonella que con su alegría siempre me apoyan y animan a continuar; al Hno. John Jiménez de la comunidad de Misioneros Oblatos quien a su momento me supo avisar y recomendar sobre esta maestría, a mi novia Lady Guissela una persona importante que me brinda su apoyo, ánimo, paciencia y cariño.

Agradezco a mis compañeros de estudio de maestría Álvaro y Juan, a quienes deseo lo mejor en sus vidas personales y profesionales, gracias por el gran grupo que formamos. De igual manera el agradecimiento para mis compañeros de estudio de Ingeniería en Electrónica y Telecomunicaciones de la EPN que han sido motivación para seguirme preparando.

Un agradecimiento especial a mi director de tesis, Dr. Gustavo Salazar por animarme al desarrollo del presente tema, que ha sido un completo aprendizaje y que como resultado ha fortalecido mis conocimientos en temas de networking, muchos éxitos en su carrera que Dios lo bendiga.

INDICE DE CONTENIDOS

INDICE DE FIGURAS

INDICE DE TABLAS

INDICE DE FÓRMULAS

1 CAPÍTULO 1 – INTRODUCCIÓN

1.1 INTRODUCCIÓN

En la actualidad el continuo acceso a Internet sea por redes de alta velocidad FTTx[1] o 5G, la demanda de contenido multimedia de gran calidad, redes sociales, VoIP, el manejo de grandes volúmenes de información a través de BigData, el uso de Internet de las Cosas (IoT), inteligencia artificial, comercio electrónico y todo tipo de Tecnologías de la Información y Comunicación TICs que la sociedad demanda en un mundo que está experimentando la transformación digital, requiere por parte de los operadores de red fijo/móviles contar con mayor capacidad de computación para brindar servicios de buena calidad, con gran estabilidad, seguridad y que sean escalables en el tiempo (Salazar Ch., Venegas, Baca, Rodríguez, & Marrone, 2018). Esto ha conllevado a que, desde los orígenes de las redes de datos las tecnologías de transporte hayan evolucionado para soportar las exigencias que en cada época se han requerido.

Con la complejidad y sobrecarga de conectividad IP y protocolos que cada tecnología ha ido introduciendo para mantener conectada a la Internet moderna, es prioritario para los Service Providers[2] buscar una simplificación de sus redes IP, con el fin que éstas sean más escalables, eficientes, simplificadas y rentables. Esto es posible a través del enrutamiento de segmentos conocido como Segment Routing. (Santos, 2019)

El desarrollo del presente trabajo trata los siguientes temas:

En el capítulo 1 se realiza la introducción al presente trabajo, justificación, antecedentes, objetivos y metodología para el desarrollo del caso de estudio.

[1] *FTTx. – Fiber to the x, representa la tecnología de acceso por fibra óptica hacia el cliente, entre los más comunes están Fiber to the Home(FTTH), Fiber to the Building(FTTB)*

[2] *SP. – Service Provider hace referencia a un operador, proveedor de servicio de Internet, Datos, IPTV, entre otros servicios, ejemplos de SP en Ecuador son cnt e.p., telconet, level 3, puntonet, netlife, claro*

El capítulo 2 presenta la situación actual y estado del arte, donde se describe la evolución en cuanto al uso de tecnología de transporte de datos preferida por los service providers hasta la actual tecnología que es MPLS con su respectiva mejora MPLS-TE (*Multi-Protocol Label Switching-Traffic Engineering*), sus bondades y desafíos que presentan en el entorno de producción y sus mejoras como Segment Routing y SD-WAN[3].

El capítulo 3 presenta el análisis del protocolo Segment Routing IPv6 como la propuesta hacia la simplificación de una red más fácil de administrar y operar; presentando sus ventajas y desventajas frente al actual MPLS, se describe en breve el concepto de QoS; también se describe el emulador de redes a utilizar en el presente trabajo, EVE-NG y las herramientas para inyección de tráfico.

En el capítulo 4 se realiza la emulación de una red en un entorno de service provider, para interconectar dos sucursales remotas, con tres escenarios que son MPLS L3VPN, SR MPLS y SRv6. Luego de lo cual se establece una comparativa de estos escenarios para determinar las ventajas de SRv6 a fin de determinar su factibilidad de implementación en service providers.

Se finaliza el trabajo con conclusiones y recomendaciones, que se extraen del desarrollo de este trabajo.

[3] *SD-WAN. – Software Define Wide Area Network hace referencia a la administración de la red WAN de forma automática mediante un controlador, con lo cual el ingeniero de redes puede administrar de manera óptima los dispositivos, también permite supervisar el estado de los enlaces WAN para su uso de acuerdo a prioridades como VoIP* (Parada Visual, 2019)

1.2 JUSTIFICACIÓN

Debido a la aparición de nuevas tecnologías como XGPON, 5G, IoT, inteligencia artificial, BigData, el procesamiento que realizan las redes IP por parte de los operadores se está sobrecargando, lo cual no va a permitir que estas redes sean escalables en un futuro (Salazar Ch., Venegas, & Marrone, 2019). Por tanto, se requiere una nueva arquitectura que sea más sencilla de administrar y operar, optimizando su automatización gracias a la adopción de las capacidades de las redes definidas por software, SDN. (Santos, 2019)

El uso de redes IP/MPLS satisface las necesidades actuales del mercado lo que se traduce en rentabilidad para los Service Providers; por las razones de crecimiento exponencial de manejo y procesamiento de información, este modelo de negocio puede verse amenazado al no poder satisfacer nuevos requerimientos de transporte de información. Es en este punto que se hace imprescindible describir las ventajas que presenta el uso del protocolo Segment Routing, al ser concebido para desarrollarse en un entorno IPv6 con soporte para IPv4 y también por la sencillez que representará para los operadores su manejo que hasta ahora requiere personal altamente calificado en administrar redes IP/MPLS.

La emulación de la red de un proveedor de servicio con Segment Routing en un entorno de producción es importante ya que va a permitir analizar y evaluar esta tecnología como la propuesta de MPLS de nueva generación, a fin de analizar el comportamiento de la red para mostrar sus ventajas y ser la tecnología a tener en cuenta para una transición futura de MPLS a Segment Routing (Salazar Ch., Naranjo, & Marrone, 2018).

Una vez que los datos demuestren las mejoras que introduce este protocolo, se puede determinar su factibilidad de implementación en el mundo real, con lo cual se establece una referencia que garantice a los operadores realizar una migración paulatina con ahorro de costes operativos y económicos.

1.3 ANTECEDENTES

Los operadores de red a menudo necesitan controlar cómo los paquetes fluyen a través de sus redes ya sea por motivos de desempeño o seguridad, así a lo largo de las décadas varias técnicas han sido propuestas para solucionar estos problemas, tales como TDM, Frame Relay, ATM, MPLS. (Duchene, Jadin, & Bonaventure, 2018)

En las redes IP tradicionales, el envío de paquetes se lo realiza en base al método *hop-by-hop* con routers independientes que tratan de establecer la mejor ruta entre un origen y destino en base al costo de dicha ruta y el flujo de paquetes tomará este camino, incluso en momentos de congestión a pesar de existir otros caminos subutilizados o inactivos. Para mejorar este inconveniente se introdujo el concepto de Ingeniería de Tráfico *(Traffic Engineering TE*) donde los routers de ingreso establecen la ruta que un paquete toma para flujos específicos, con lo cual en situaciones de saturación, en lugar de tomar la ruta de menor costo, el paquete utiliza aquellas rutas no utilizadas permitiendo de esta manera un balance de carga. Junto a esto actúa el protocolo de señalización RSVP (*Resource Reservation Protocol*) que permite reservar recursos para un tráfico específico, dando lugar a MPLS-TE que permite configurar una ruta de conmutación de etiquetas con la tecnología RSVP-TE.

El problema radica en entornos de service provider ya que el protocolo LDP (*Label Distribution Protocol*) y RSVP-TE (*Resource Reservation Protocol-TE*) se vuelven complejos de implementar, mantener, operar y solucionar; por el hecho que generan mucho tráfico de señalización, tienen una comprensión limitada de la topología e inundan la red de túneles MPLS. Esto obliga a los operadores a contar con personal especializado que brinde soporte a esta arquitectura. (Santos, 2019)

Frente a esto, Segment Routing se presenta como una solución flexible y escalable de implementar, ya que el router origen elige una ruta y la inserta en la cabecera de un paquete como una lista de direcciones, así ya no se realiza el proceso de *hop-by-hop* y se utiliza segmentos para el envío, lo que convierte a la red más fácil de operar.

Todavía, en los operadores de red hay procesos que se realizan por nodo, lo que se traduce en gastos operativos, tiempo, dinero; frente a esto Segment Routing IPv6 simplifica la administración de sus redes a través de SRN (*Software Resolved Network*) mediante el uso de un controlador que administra recursos de la red, donde las aplicaciones pueden interactuar con el controlador para indicar sus requerimientos de flujos. (Duchene, Jadin, & Bonaventure, 2018)

1.4 OBJETIVOS

1.4.1 Objetivo General

Emular y evaluar una red con Segment Routing IPv6 para determinar su factibilidad de implementación en service providers.

1.4.2 Objetivos Específicos

- Describir la tecnología actual de transporte utilizada por service providers.
- Estudiar el protocolo Segment Routing IPv6 con características y ventajas frente a tecnologías actuales.
- Emular una red con el uso del protocolo Segment Routing IPv6, para interconectar dos sucursales.
- Establecer un cuadro comparativo entre MPLS, SR MPLS, SRV6 explicando sus ventajas y desventajas.

1.5 METODOLOGÍA

Para describir la tecnología actual utilizada por service providers se exponen conceptos básicos de MPLS, arquitecturas de operación, características y desafíos que deben cumplir a futuro para redes de nueva generación.

El marco teórico describe el protocolo Segment Routing IPv6, para lo cual se estudia su teoría, funcionamiento, ventajas y desventajas. Se describe el concepto de Calidad de Servicio y sus parámetros principales. También se presenta las herramientas para emulación de redes (EVE-NG, 2019) e inyección de tráfico (iPerf, 2019)

Para emular una red con Segment Routing IPv6, se va a trabajar con una topología de Service Provider genérica (conexión Matriz-Sucursal empresarial), en la cual se va a interconectar dos sucursales y que como antecedente también se emulará las tecnologías de MPLS y SR MPLS (SR-IPv4); estas topologías se van a implementar en el emulador de redes EVE-NG.

Para realizar toma y evaluación de datos, se utilizarán comandos propios de los equipos Cisco IOS XR reales, emulados mediante Dynamips, y así extraer información útil para el análisis de esta tecnología disruptiva en el área de Service Providers.

En base a los resultados obtenidos se va a determinar la factibilidad de implementación de Segment Routing IPv6 en redes de Service Provider frente a tecnologías actuales como MPLS-LDP.

2 CAPÍTULO 2 - SITUACIÓN ACTUAL Y ESTADO DEL ARTE

2.1 Estado del Arte

A pesar que en la actualidad las tecnologías soportan las demandas de IP en cuanto a flexibilidad y escalabilidad, los operadores se están anticipando al gran volumen que se va a experimentar por servicios como cloud-based, o aquellos con altos SLA[4], por lo que se presenta a Segment Routing SR como la arquitectura de red que cubre ese hueco mediante la implementación de enrutamiento de origen y paradigmas de tunneling, permitiendo a los nodos dirigir paquetes sobre rutas, usando una secuencia de instrucciones(segmento) que se coloca en el encabezado del paquete. (Filsfils, Kumar Nainar, Pignataro, Cardona, & Francois, 2015)

De acuerdo a la (Internet Engineering Task Force (IETF), 2018) en la RFC8402, se indica que Segment Routing SR se basa en el hecho que un nodo origen encamina un paquete mediante una lista ordenada de instrucciones que define una ruta topológica específica que se llama "segmento", y que se puede aplicar sobre la arquitectura MPLS, ya que los segmentos se tratan como una etiqueta MPLS que se codifican en la parte superior del stack de etiquetas, por tanto, al completar un segmento, la etiqueta que lo relaciona se saca del stack. Segment Routing se puede implementar en IPv6, donde la lista de segmentos se codifica como una lista ordenada de direcciones IPv6 en el encabezado de enrutamiento; un segmento activo se indica por el campo Destination Address (DA) del paquete.

Se conoce que en MPLS se realizan los procesos de PUSH, SWAP, POP, a lo largo de una ruta en la cual las etiquetas cambian de acuerdo a cada salto de router, en tanto que Segment Routing establece una ruta desde el inicio para un paquete, lo que permite mejoras al proceso

[4] *SLA. - Service Level Agreement, es el acuerdo de nivel de servicio establecido entre el proveedor de servicio y el cliente, en cuanto a disponibilidad de servicio, por ejemplo 99.9%*

de MPLS, como se lo explica en el video de CISCO, titulado *Introduction to Cisco Segment Routing Traffic Engineering*. (CISCO, 2018).

Actualmente en el Ecuador y en su gran mayoría en el resto de países de nuestra región, los proveedores de servicio tienen implementada su arquitectura de core y agregación en base a MPLS que frente a nuevas demandas de tráfico generadas por las redes cableadas como FTTH, o inalámbricas como 4G LTE, y en un futuro cercano 5G, demandan de MPLS una mejor respuesta para adaptarse a nuevos patrones de tráfico con mejores respuestas de latencia y que permitan tener conectividad MPLS extremo a extremo, como el caso de los e-node B, o equipos de acceso en la redes fijas. En julio de 2019 se realiza la investigación "Diseño de una red IP MPLS utilizando la arquitectura Seamless para un proveedor de servicios de telecomunicaciones con cobertura en la región 3 de Ecuador", en donde se propone el diseño de una arquitectura Seamless MPLS para poder segmentar de forma lógica y jerarquizar la red para escalar en el número de nodos con robustez, simplicidad y mejores prestaciones, para satisfacer demandas actuales de tráfico en las provincias que conforman la Región 3 como son Tungurahua, Pastaza, Cotopaxi y Chimborazo (Vinueza, 2019).

2.2 Requerimientos y demandas en redes de nueva generación

Como se expone en la introducción, el mundo de las TICs experimenta un crecimiento acelerado debido al acceso a contenido e información que cada vez aumenta su volumen, siendo este acceso a través de redes de alta velocidad como FTTx o 5G (Naranjo & Salazar Ch., 2017). Por tanto la capacidad de cómputo y procesamiento de información requiere que los operadores en su core de la red puedan gestionar de manera eficiente estos grandes volúmenes de información, sin degradar su velocidad de respuesta y calidad de servicio; esto actualmente se consigue gracias al uso de MPLS que mediante la conmutación de etiquetas mantiene la velocidad de procesamiento, pero como se menciona al inicio de este párrafo, cada vez la información está creciendo y en grandes cantidades, lo que exige de los

operadores contar con nuevas técnicas para poder manejar estas cargas sin afectar la experiencia del cliente. Es por tanto que Segment Routing suple estas necesidades al permitir que una red sea más rápida, escalable, simple de administrar, fácil para implementar políticas de ingeniería de tráfico y sea rentable (Salazar Ch., Naranjo, & Marrone, 2018).

2.3 Evolución de tecnologías hasta situación actual de Service Providers

La evolución de las tecnologías para transporte de datos tiene sus inicios desde el desarrollo de la informática en los años 60's, junto con las bases de la Teoría de la Información que han tenido gran aporte por parte de Claude Shannon, Harry Nyquist y Ralph Hartley a inicios del siglo XX.

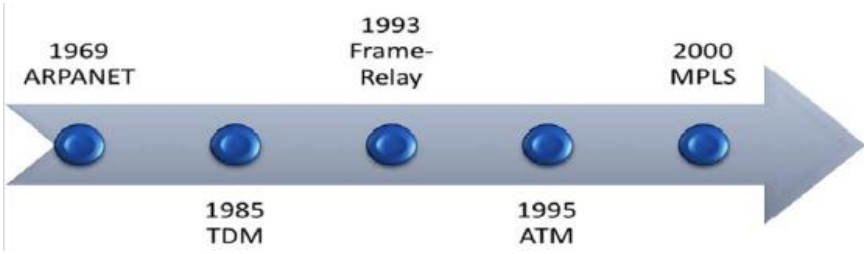


Figura 1. Evolución de tecnologías WAN
Obtenido de (Salazar Ch., Naranjo, & Marrone, 2018)

ARPANET, creado en DARPA (*Defense Advanced Research Projects Agency)* que en diciembre de 1969 permite enviar el primer mensaje desde un computador de Leonard Kleinrock en UCLA (*Universidad de Los Ángeles California*) a otro ubicado en SRI *(Instituto de Investigaciones Standford),* a través de una línea telefónica a baja velocidad.

Figura 2. Inicio de Arpanet 1969
Obtenido de (juuncaal, 2019)

Junto con esto en 1975 surge el protocolo TCP/IP creado por Vinton Cerf y Robert Kahn, y que permite comunicar dos computadores con diferente sistema operativo. (Salazar, Comunicaciones Unificadas y VoIP - PUCE, 2019)

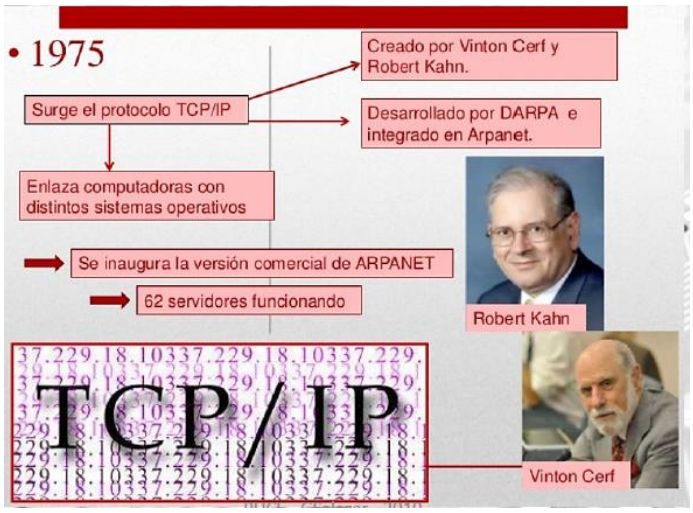


Figura 3. Protocolo TCP/IP
Obtenido de (Salazar, Comunicaciones Unificadas y VoIP - PUCE, 2019)

Con la aparición de las redes WAN cuyo objetivo es comunicar nodos distantes, se empiezan a desarrollar tecnologías que traen mejoras a su predecesora.

Redes Orientadas a Circuitos es la forma tradicional de funcionamiento de las redes telefónicas, donde es necesario establecer un circuito entre los extremos antes de iniciar la

comunicación, siendo este circuito único, que no puede utilizarse por otros y que se libera una vez que se termina la conversación. La calidad del enlace es constante al ser un camino físico con la desventaja que resulta caro ya que requiere un circuito por cada conversación y es poco escalable.

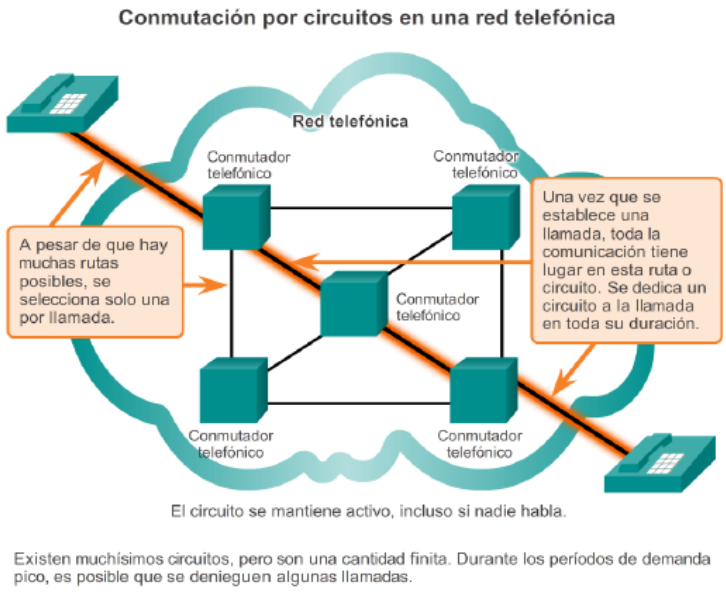


Figura 4. Conmutación de circuitos
Obtenido de (Salazar, Comunicaciones Unificadas y VoIP - PUCE, 2019)

Redes Orientadas a Paquetes basa su nombre en el hecho que, flujos de tráfico de información diferente se fragmentan en paquetes y viajan por una o diferentes rutas y medios de transmisión. Al llegar al otro extremo los paquetes se vuelven a unir para obtener el mensaje original. A pesar que la red puede seguir aceptando paquetes, la velocidad puede ser lenta, por lo cual se puede manejar prioridades, lo que da lugar a la aparición de colas.

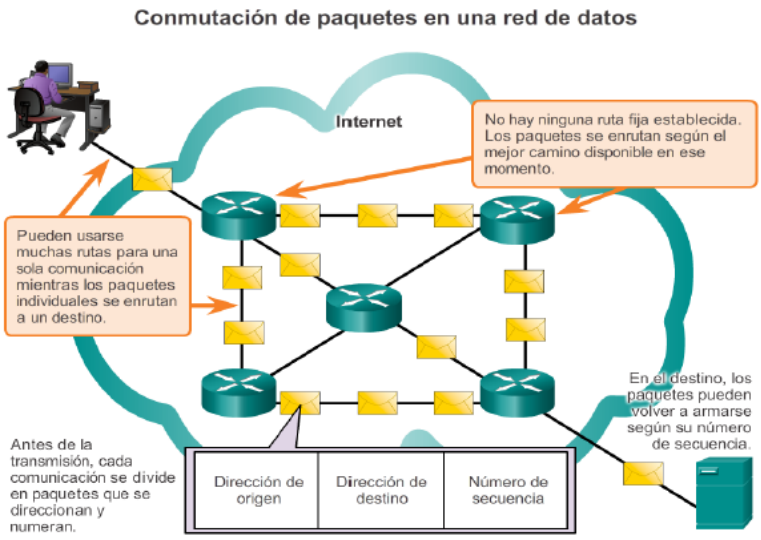


Figura 5. Conmutación de paquetes
Obtenido de (Salazar, Comunicaciones Unificadas y VoIP - PUCE, 2019)

TDM *Time Division Multiplexing* permite a un sistema de transmisión asignar el ancho de banda total a un canal durante una fracción del tiempo. Se conoce que el rango de la voz humana está entre los 400 a 4Khz. Por el teorema de Nyquist, la frecuencia de muestreo de una señal debe ser el doble del ancho de banda original.

$$f_s = 2f_{max} \rightarrow f_s = 8[KHz]$$

Fórmula 1. Teorema de Nyquist

Esto significa que el intervalo de tiempo entre cada muestreo es el inverso de la frecuencia que da como resultado

$$T_s = \frac{1}{f_s} = \frac{1}{8[KHz]} = 125[useg]$$

Fórmula 2. Tiempo de muestreo para voz humana

También se conoce que para codificar cada muestra se utilizan 8 bits dando lugar a M = 2^8 = 256 niveles. Con lo expuesto se tiene que la velocidad de una señal o canal telefónico corresponde a

$$V_{tx} = 8000 \cdot 8 \left[\frac{bits}{seg}\right] = 64[kbps]$$

Fórmula 3. Velocidad de transmisión de un canal telefónico

Gracias a TDM es posible multiplexar varias llamadas telefónicas en los enlaces troncales entre centrales telefónicas.

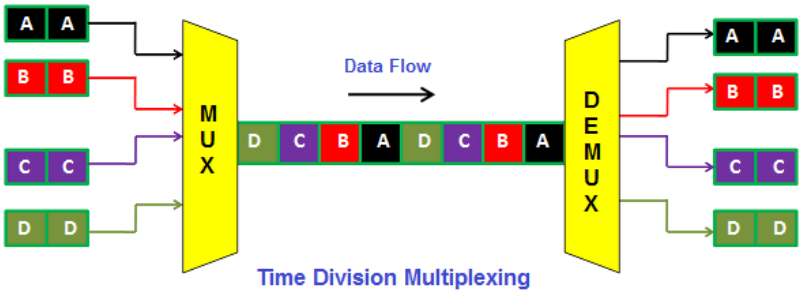


Figura 6. Esquema de Time Division Multiplexing
Obtenido de (Practonet)

PDH *Plesiochronous Digital Hierarchy* Es la tecnología que permite asociar varios canales telefónicos y enviarlos por un mismo medio de transmisión sea cableado como coaxial, cobre, fibra óptica o inalámbrico como microonda. Se consigue agrupar los canales a través de la técnica de TDM. Existen dos estándares de agrupación que son:

- Europeo: agrupa 30 canales de datos + 2 canales de señalización con lo cual se tiene

$$E1 = 32 \cdot 64\ [kbps] = 2.048[Mbps]$$

Fórmula 4. Velocidad de E1

- Norteamericano: agrupa 23 canales de datos + 1 canal de señalización que dan aproximadamente

$$T1 = (24 \cdot 64 + 8)[kbps] = 1.544[Mbps]$$

Fórmula 5. Velocidad de T1

En PDH a esta agrupación muy utilizada por proveedores de servicio en su momento con la aparición de las redes ISDN (*Integrated Services Digital Network*), se la conoce como E1/T1 PRI.

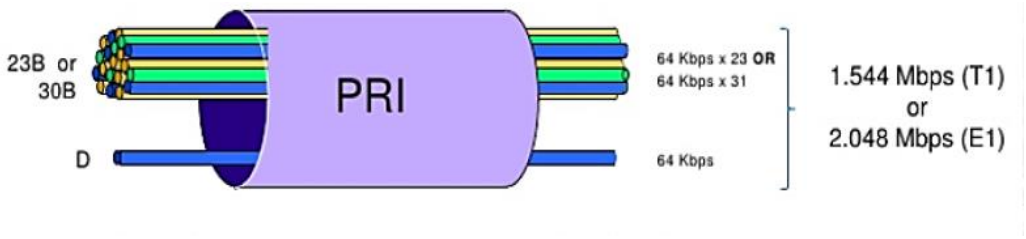


Figura 7. E1/T1 PRI
Obtenido de (Salazar, Comunicaciones Unificadas y VoIP - PUCE, 2019)

Dado que en Latinoamérica se trabaja con el estándar europeo, para obtener las jerarquías superiores se multiplica por 4 a partir de la siguiente jerarquía:

Tabla 1. *Velocidades para jerarquía PDH Europeo*

JERARQUIA	VELOCIDAD DE TX
E1	2.048 Mbps
E2	8.192 Mbps
E3	32.768 Mbps
E4	131.072 Mbps

Fuente: (Universidad Publica de Navarra)

SDH / SONET Synchronous Digital Hierarchy / Synchronous Optical Network Al igual que PDH permite transportar información a mayores velocidades que están sincronizadas en toda la red, permitiendo añadir y extraer señales o tributarios de menores velocidades, por su arquitectura de conexión que tiende a formar anillos presenta alta disponibilidad frente a fallos lo cual garantiza que la información esté disponible. A continuación se presentan las velocidades de acuerdo a las jerarquías tanto en SDH como en SONET.

Tabla 2. *Velocidades para jerarquías SDH/SONET*

JERARQUIA		VELOCIDAD DE TX
SDH	SONET	
	OC-1	51.84 Mbps
STM-1	OC-3	155.52 Mbps
STM-4	OC-12	622.08 Mbps
STM-16	OC-48	2488.32 Mbps ≈ 2.5 Gbps
STM-64	OC-192	9953.28 Mbps ≈ 10 Gbps
STM-256	OC-768	39813.12 Mbps ≈ 40 Gbps

Fuente: (Universidad Publica de Navarra)

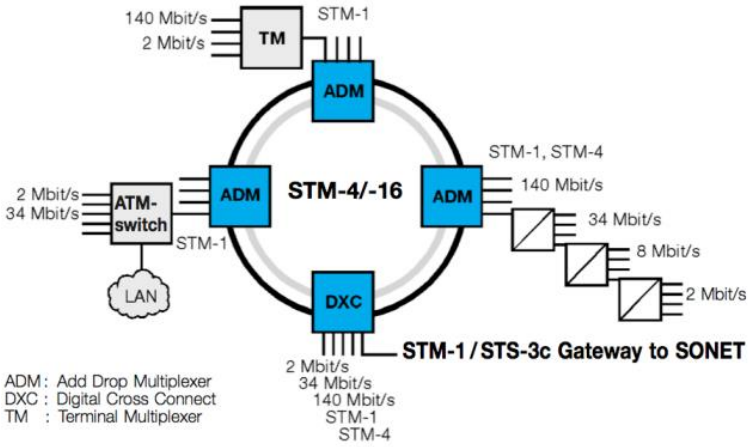


Figura 8. Topología de una red SDH
Obtenido de (Universidad Publica de Navarra)

DWDM Dense Wavelenght Division Multiplexing que no se aborda a profundidad, pero se describe como la multiplexación de señales a gran velocidad espaciados por longitud de onda entre dos canales adyacentes, actualmente DWDM permite transportar 40, 80, 160 longitudes de onda de 10Gbps de capacidad, con un espaciamiento promedio de 0.8/0.4 [nm] (Don, 2018). Con estas capacidades se puede hacer frente a las grandes demandas de tráfico actuales que el mundo de las TICs exige.

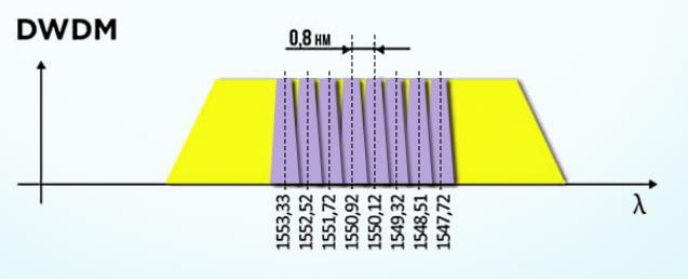
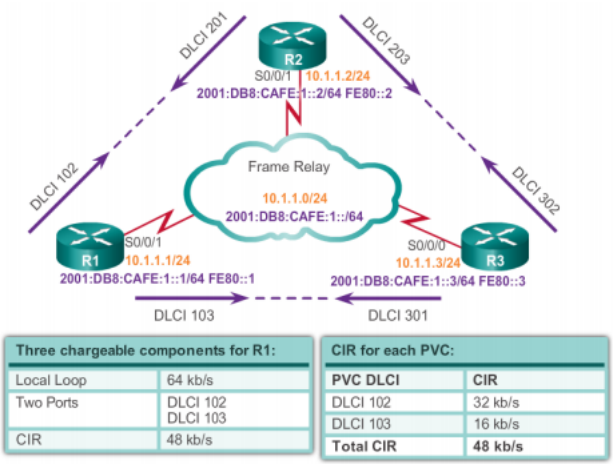


Figura 9. Distribución de longitud de onda DWDM
Obtenido de (Don, 2018)

Frame Relay se presenta por el año 1984, siendo adoptado a finales de los años 80 debido a su falta de interoperabilidad, tomando fuerza en los años 90 cuando las grandes compañías como Cisco, Digital Equipment, Northern Telecom, Stratacom, Convex Computer, invierten en su desarrollo. Se muestra como una tecnología de conmutación orientada a paquetes de datos, que emulan circuitos virtuales definidos por software donde se establece un trayecto

privado entre dos nodos, con velocidades iniciales de 2Mbps y que pueden ser escalables. Existen dos clases de circuitos virtuales que son PVC (*Permanent Virtual Circuit*) y SVC (*Switched Virtual Circuit*) y que con mayor detalle se exponen en el numeral 2.4.1



Three chargeable components for R1:	
Local Loop	64 kb/s
Two Ports	DLCI 102 DLCI 103
CIR	48 kb/s

CIR for each PVC:	
PVC DLCI	**CIR**
DLCI 102	32 kb/s
DLCI 103	16 kb/s
Total CIR	**48 kb/s**

Figura 10. Diagrama de red de Frame Relay
Obtenido de (Cisco Networking Academy, 2014)

ATM Asynchronous Transfer Mode Es la arquitectura de red basada en celdas, en lugar de la arquitectura basada en tramas. Estas celdas tienen una longitud de 53 bytes, que contienen 5 bytes de cabecera + 48 bytes de datos. Maneja el concepto de VPI (*Virtual Path Identifier*) y VCI (*Virtual Circuit Identifier*) que identifica al circuito.

Se define con el estándar ITU-T I.150, realiza la conmutación de paquetes orientado a conexión mediante la emulación de circuitos virtuales, garantizando capacidad y retardo constante para tráficos de voz, video y datos. Su rapidez reside en el hecho de realizar conmutación de celdas que son de tamaño pequeño

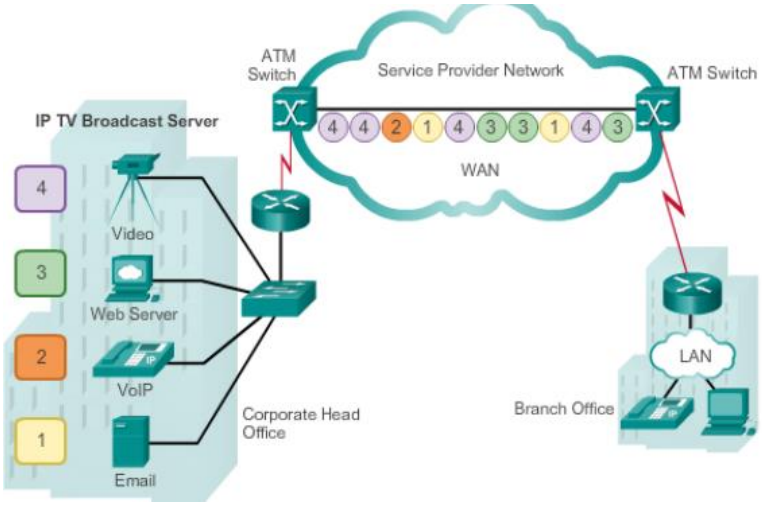


Figura 11. Arquitectura ATM
Obtenido de (Cisco Networking Academy, 2014)

Al igual que en Frame Relay, se pueden crear los siguientes tipos de circuitos en ATM:

- PVC (*Permanent Virtual Circuit*) que se realiza de forma manual, es más fácil de depurar, no es escalable.
- SVC (*Switched Virtual Circuit*) de forma dinámica se establece mediante señalización, presenta pronta recuperación frente a fallos de los enlaces, es más complejo que PVC.
- Soft-PVC realiza configuración manual en los extremos y SVC en el core de la red.

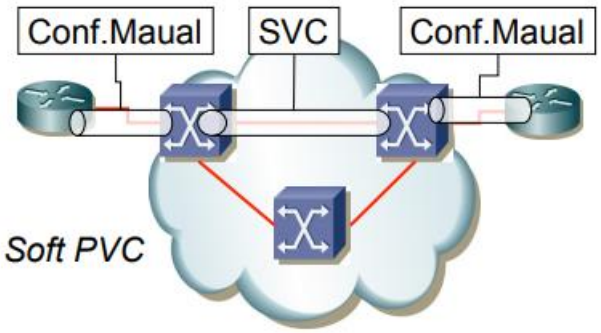


Figura 12. Conexión Soft PVC en ATM
Obtenido de (Universidad Publica de Navarra)

MPLS Multiprotocol Label Switching Es el método actual usado por los proveedores de internet para envío de paquetes, ya que es una tecnología WAN de alto desempeño que realiza el envío de datos de un router a otro basado en una ruta de etiquetas o *labels* como se aprecia en la Figura 13, en lugar de hacerlo a través de direcciones de red IP y búsqueda de rutas de forma tradicional, permitiendo servicios como:

- Acceso a Internet
- VPN *Virtual Private Network*
- Telefonía
- QoS *Quality of Service*

Actualmente se usa MPLS, ya que los datos se transforman sobre la arquitectura IP, así todo corre sobre IP, Ethernet reemplaza a ATM+SDH, con lo cual IP/MPLS es el core actual de los service providers. MPLS provee un encapsulamiento intermedio entre los headers de capa 3 y capa 2 del modelo OSI, junto con el hecho de realizar los envíos basados en labels en lugar del payload de un paquete, con lo cual diferentes protocolos pueden ser usados para determinar una ruta, diferentes payloads pueden ser usados para proveer diferentes servicios y cualquier servicio tradicional puede ser implementado en un entorno MPLS.

MPLS añade un header al paquete que se lo conoce como MPLS label; los labels hacen referencia a la red IP de destino, así cada destino tiene una label que le corresponde en cada router MPLS. Los beneficios de usar MPLS son:

- Reduce sobrecargas de reenvío en routers de core
- Soporta el envío de protocolos no-IP
- Mejora el enrutamiento BGP
- Soporta múltiples aplicaciones como enrutamiento unicast/multicast, VPN, TE, QoS, AToM(*Any Transporto over MPLS*)

La distribución de etiquetas se lleva a cabo gracias al protocolo LDP (Label Distribution Protocol), donde los LSR (Label Switching Router) comparten información para poder alcanzar otros LSR. Con el uso del protocolo RSVP (Resource *Reservation Protocol*) se puede reservar recursos en toda la ruta de un LSP (*Label Switching* Path) para un flujo de tráfico específico (CISCO, 2014)

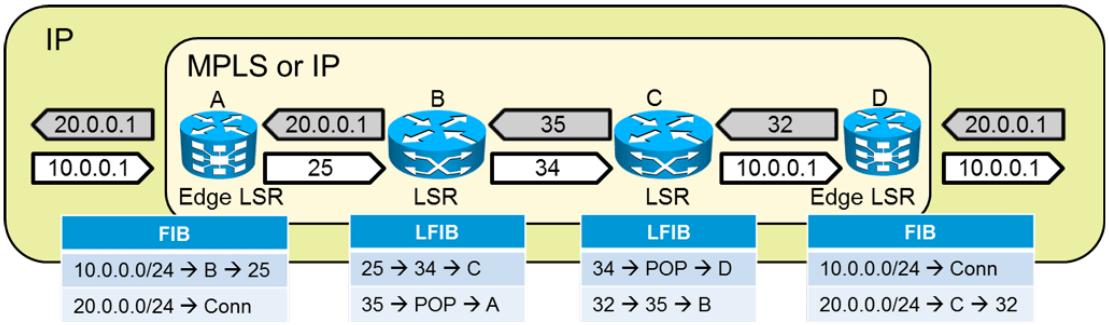


Figura 13. Arquitectura MPLS
Obtenido de (CISCO, 2014)

SR SEGMENT ROUTING: A diferencia de MPLS tradicional, y las complicaciones que exige el operar una red, Segment Routing o SR, permite implementar el enrutamiento IP de forma flexible y escalable, así el router de origen elige una ruta y la inserta como una lista ordenada de enlaces, con lo cual esta ruta no depende de señalizaciones *hop-by-hop*, LPD, RSVP-TE, de ahí el hecho que usa *segmentos* para hacer un envío. SR no es una nueva tecnología frente a su antecesora MPLS, más bien puede operar sobre un data plane de MPLS o IPv6, junto a sus servicios como L3VPN. La implementación de SR en los proveedores actuales depende en el hecho de contar con una plataforma de software de automatización inteligente que permita la evolución de la red a operar como se muestra en la Figura 14. (Santos, 2019).

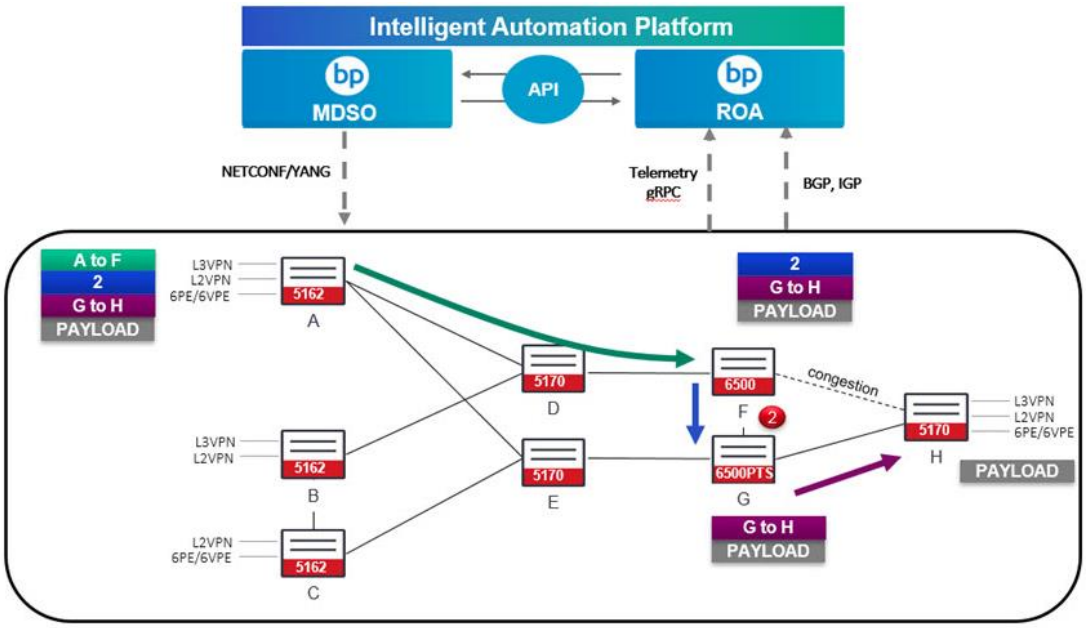


Figura 14. Segment Routing - implementación de adaptive IP
Obtenido de (Santos, 2019)

Los dos puntos claves que Segment Routing introduce como mejora y que combina lo mejor de MPLS y Source Routing es el poder codificar la ruta explícita en el paquete en el router de ingreso y, poder codificar la información de esa ruta explícita en paquetes etiquetados de tal manera que la red MPLS pueda procesarlos sin necesidad de almacenar estados adicionales en los routers a lo largo de la ruta deseada. Segment Routing afianza la idea de que ***una ruta explícita es un conjunto ordenado de instrucciones puestos en el paquete***, con los routers ejecutando estas instrucciones a medida que los envían. A cada instrucción se la llama segmento, y tiene su propio número llamado Segment ID (SID) y que en MPLS se codifican como un stack de etiquetas, con cada etiqueta representando un segmento en particular, así los valores de etiquetas MPLS pueden llevar los Segment IDs de segmentos individuales. (The Cisco Learning Network, 2018).

En el siguiente capítulo se expone en mayor detalle la operación de Segment Routing, ya que es el tema principal de este trabajo.

SD-WAN Basa su nombre en el hecho de incorporar tecnologías SDN (*Software Defined Network*) y NFV (*Network Function Virtualization*) a una red WAN; es decir que por ejemplo el proceso de creación de túneles WAN sea rápido desde una API(*Application Programming Interface*), lo cual se muestra fácil y confiable para el operador de red, de manera que la red WAN tenga una arquitectura estandarizada, flexible, escalable y de bajo costo frente a nuevas adaptaciones que el mercado demande, contribuyendo a la optimización de todos los modelos operativos.

Funcionalidades del entorno WAN que se tienen con SD-WAN son las siguientes:

- *Gestión de políticas*, que permite configurar y gestionar parámetros como QoS, velocidades, seguridad, accesos, entre otros
- *Red por aplicación*, donde se definen redes virtuales independientes de la parte física, y por ende se pueden aplicar políticas por cada red virtual, por ejemplo una red para proveedores con acceso a una aplicación específica y en un horario determinado.
- *Asignación dinámica de servicios*, que permite añadir servicios en flujos de tráfico por un tiempo dado, como re direccionar el tráficos a un elemento en la red o a un data center. (Naranjo & Salazar Ch., 2017)
- *Dual uplink*, lo que permite utilizar el enlace activo y de respaldo de acuerdo a los servicios o flujos de tráficos.
- *Cloud Híbrida*, que permite extender las capacidades de TI a un data center de manera dinámica, sencilla y transparente.
- *Auto petición/autogestión,* mediante un sitio web el cliente puede ejecutar o solicitar políticas, cambios, monitoreo, reportes de la WAN

- *Monitorización e informes*, como se ha indicado a través del portal de clientes, se puede obtener información en tiempo real de la configuración, topología de red, tráficos, troubleshooting, y aquello que el cliente considere necesario. (Capillas, 2016)

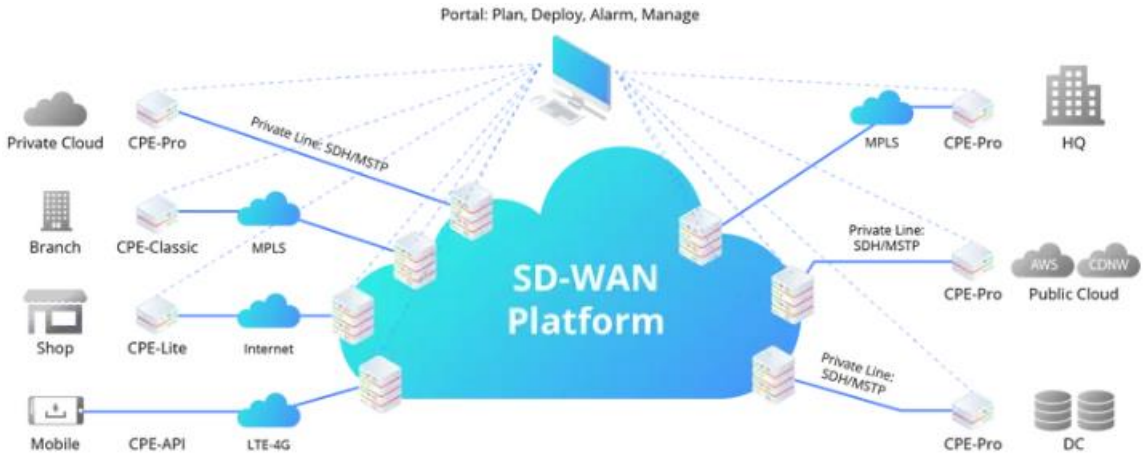


Figura 15. Arquitectura SD-WAN
Obtenido de (Parada Visual, 2019)

2.4 Descripción de tecnología de envío de paquetes usada por Service Providers

En este numeral se presenta en breve las tecnologías actuales para envío de paquetes utilizadas por los proveedores de servicio.

2.4.1 Frame Relay

Esta tecnología WAN muy popular en los años 90, muy poco usada en la actualidad, merece su explicación y comprensión dentro de las tecnologías WAN, ya que permite comunicaciones entre todos los sitios remotos utilizando un solo enlace hacia el proveedor de servicios, como se muestra en la Figura 16.

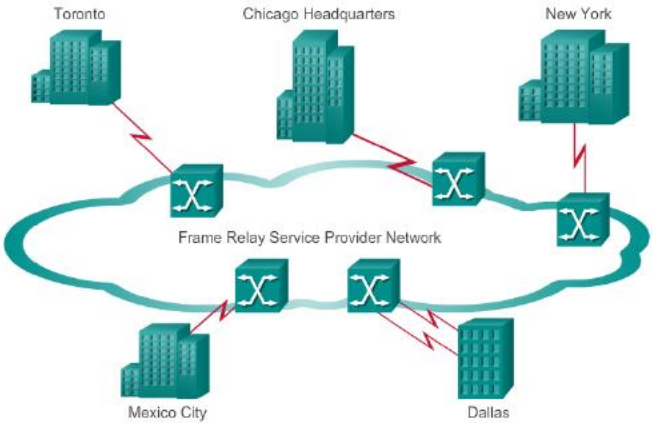


Figura 16. Red Frame Relay proveedor – cliente
Obtenido de (Cisco Networking Academy, 2014)

Frame Relay introduce el concepto de dos tipos de circuitos que son:

- *Permanent Virtual Circuit (PVC),* que se define por el administrador de red y cuyo recurso es exclusivo para el servicio que ha sido creado incluso en caso de existir cambios en la red donde a través de enrutamientos automáticos se conserva el circuito creado lo que emula a un circuito dedicado punto a punto, pero a menor costo que uno real.
- *Switched Virtual Circuit (SVC),* son circuitos que se establecen dinámicamente mediante el envío de mensajes de señalización, se adapta a las demandas de red y se forma cuando un determinado servicio entre dos o más nodos es requerido. (TELCEL BELLSOUTH)

Los VCs se identifica a través de un DLCI (*Data Link Connection Identifier*), que tienen un significado local, y cuyos valores se pueden repetir en toda la red WAN, por tanto un DLCI no tiene significado más allá del enlace local. Frame Relay garantiza la comunicación bidireccional entre dispositivos. (Cisco Networking Academy, 2014)

Las topologías básicas de operación en Frame Relay son: Star-Hub y Mesh como se aprecia en la Figura 17.

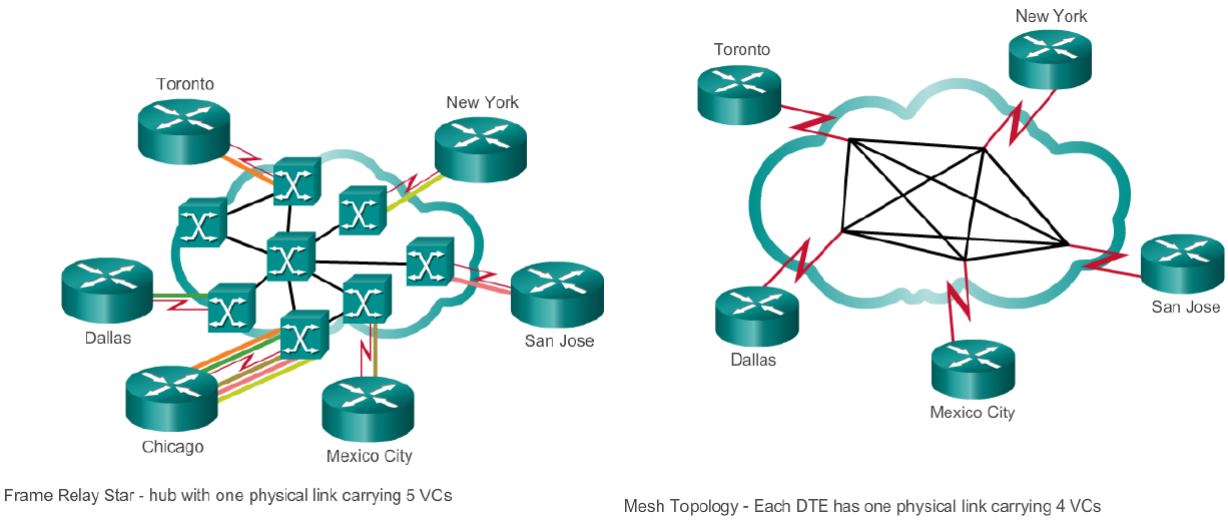


Figura 17. Topologías Frame Relay
Obtenido de (Cisco Networking Academy, 2014)

La encapsulación de la trama Frame Relay es la que se aprecia en la Figura 18, y donde se puede apreciar en que parte se incluye el DLCI, así como el FECN y BECN, muy útiles para control de flujo que se pueden apreciar en la Figura 19.

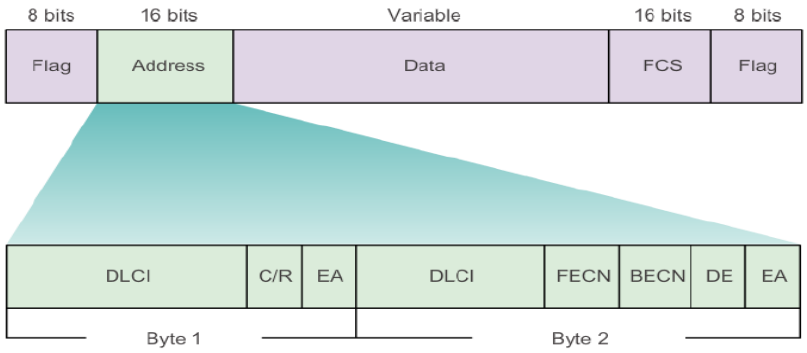


Figura 18. Encapsulación Frame Relay
Obtenido de (Cisco Networking Academy, 2014)

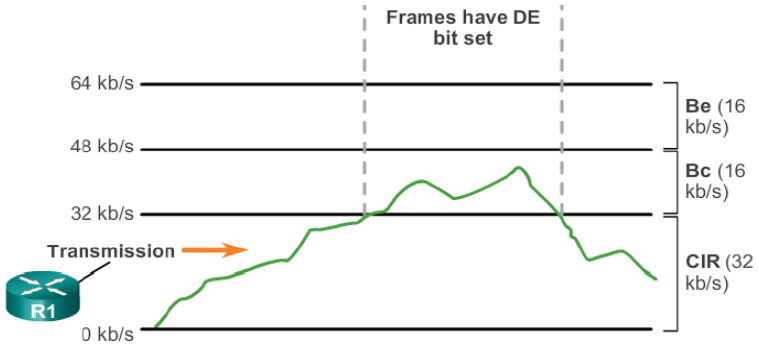


Figura 19. Bursting en Frame Relay
Obtenido de (Cisco Networking Academy, 2014)

2.4.2 **MPLS**

La conmutación multiprotocolo por etiquetas permite el transporte de datos de varios protocolos incluidos aquellos no IP, soporta ATM, Frame Relay y Ethernet, su uso es muy común para circuitos virtuales en las redes IP, obtiene lo mejor de las capas 2 y 3, para lo cual se habilita el envío de paquetes a través de LSP (*Label Switching* Path) que es una secuencia de etiquetas para alcanzar una red de destino, cabe indicar que MPLS no reemplaza el enrutamiento IP. Las aplicaciones en las que se usa esta tecnología dentro de la red de un proveedor de servicios son:

- Enrutamiento Unicast IP
- Enrutamiento Multicast IP
- MPLS TE (*Traffic Engineering*)
- QoS
- MPLS VPN
    - MPLS VPNs capa 2
    - MPLS VPNs capa 3
- AToM(*Any Transport over MPLS*) que es la solución para transportar paquetes capa 2 sobre el backbone IP/MPLS

Entre las características que MPLS usa para el envío se tienen:

- MPLS mejora el enrutamiento IP y la conmutación CEF (*Cisco Express Forwarding*) en el core de un proveedor de servicio.
- La conmutación se basa en etiquetas, las mismas que por lo general corresponden a las redes de destino IP.
- Sólo los routers de frontera o edge, realizan la búsqueda de ruta o routing lookup.

- Una cabecera o header adicional llamada etiqueta MPLS es insertada y se usa para la conmutación MPLS.

Es conveniente comprender la terminología que se utiliza en el dominio MPLS, por lo cual se definen tres tipos de routers:

- **Ingress LSR:** Es el router de frontera de una red MPLS y es el primero en insertar un header MPLS y una etiqueta al paquete.
- **Egress LSR:** Es el router ubicado en la frontera de la red MPLS y es el último punto antes de que el paquete abandone la red MPLS, y remueve todas las etiquetas y cabeceras MPLS.
- **Intermediate LSR:** Son los LSR que forman la red MPLS y que realizan las acciones de PUSH, SWAP, POP con las etiquetas basadas en el enrutamiento con MPLS. (CISCO, 2014)

Los equipos Ingress LSR y Egress LSR se los conoce como PE (*Provider Edge*), y a los Intermediate LSR se los conoce como P router (*Provider Router*), que se los aprecia en la Figura 20.

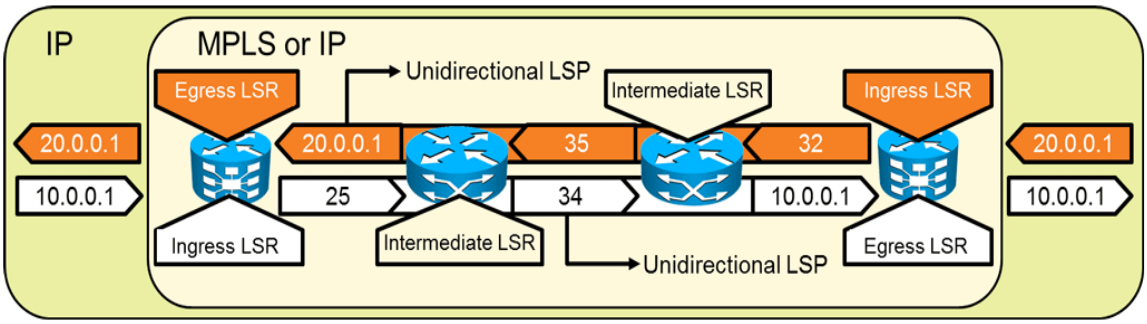


Figura 20. Equipos dentro de una red MPLS
Obtenido de (CISCO, 2014)

En la arquitectura MPLS, se definen dos planos de envío de datos que son:

- **Control Plane,** construye la tabla de enrutamiento RIB (*Routing Information Base*), y donde los protocolos de capa 3 se usan para administrar enrutamiento capa 3 como OSPF, IGRP, EIGRP, IS-SIS, RIP, BGP. Usa un protocolo de intercambio de etiquetas LDP (*Label Distribution Protocol*), que añade etiquetas a redes que son aprendidas a través de un protocolo de enrutamiento; para la versión mejorada que es MPLS-TE se usa el protocolo RSVP(*Resource Reservation Protocol*)

- **Data Plane,** se encarga del envío, basado en la dirección de destino o etiqueta, por tanto es independiente del tipo de protocolo de enrutamiento o de intercambio de etiqueta, y realiza el envío de paquetes a la interface específica basado en la información de las tablas FIB (*Forwarding Information Base)* o LFIB (*Label Forwarding Information Base*), que se construyen a partir de las de la información de RIB y del protocolo de intercambio de etiquetas respectivamente. Al data plane se lo conoce como forwarding plane. (CISCO, 2014)

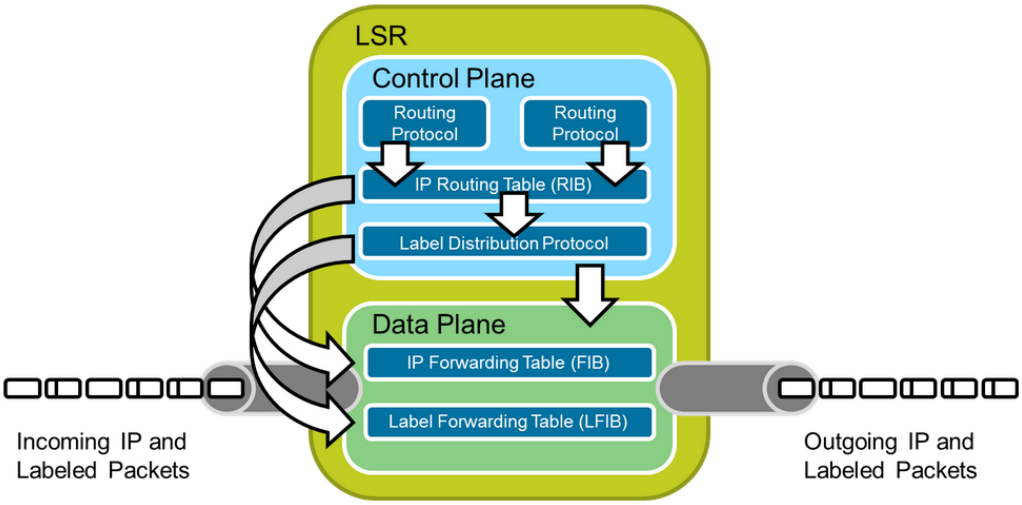


Figura 21. Control & Data Plane MPLS
Obtenido de (CISCO, 2014)

En una red MPLS, el uso de las tablas de data plane tienen las siguientes aplicaciones:

- FIB se usa para el envío de paquetes IP sin etiquetas o para etiquetar paquetes si el siguiente salto está disponible
- LFIB se usa para enviar paquetes etiquetados, luego la etiqueta se cambia (*swap*) por el siguiente salto.

El proceso de envío y recepción de un paquete de una red origen a una red de destino que atraviesa por el core MPLS de un proveedor de servicio es el que se muestra a continuación:

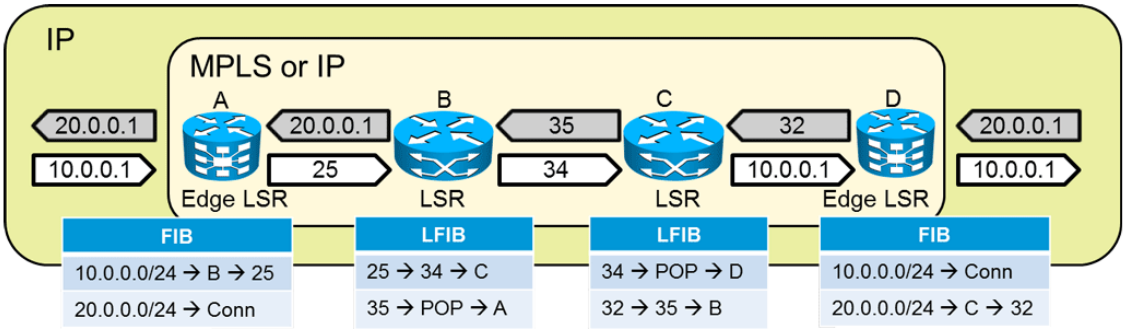


Figura 22. Envío de paquetes en una red MPLS
Obtenido de (CISCO, 2014)

La etiqueta MPLS utiliza un header o cabecera de 4bytes o 32 bits que se inserta entre las capas 2 y 3, con los siguientes detalles:

- **20 bits label**, que es la etiqueta utilizada para conmutación, se reservan los valores de 0 a 15.
- **3 bit EXP field,** usado por Cisco para definir una CoS (*Class of Service*), o IP precedence utilizada en el byte de TOS para asignar precedencia a un paquete IP.
- **1 bit Bottom-of-stack,** determina si la etiqueta es la última en el paquete, cuando su valor es (1L).
- **8 bit TTL field,** tiene el mismo significado que el TTL en un paquete IP.

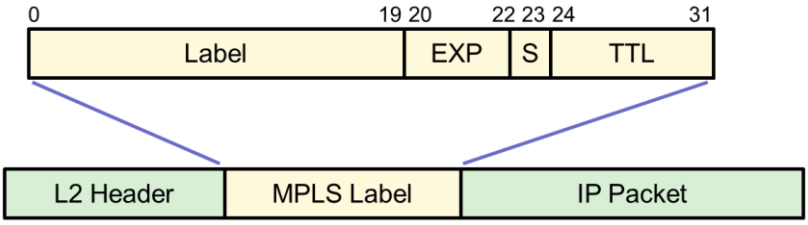


Figura 23. MPLS label
Obtenido de (CISCO, 2014)

2.4.3 **Segment Routing**

Siendo Segment Routing la más reciente tecnología en cambiar la forma en que los paquetes son manejados en infraestructuras de red críticas como en el core de Internet, data centers o entre data centers, esta propuesta es una forma más simple de controlar redes, programar enrutamiento y procesamiento de paquetes, e implementar políticas de ingeniería de tráfico. La arquitectura de Segment Routing está especificada por la IETF[5] (RFC 8402), pero su implementación en data plane está hecha sobre dos tecnologías que son MPLS e IPv6.

El data plane de MPLS permite llevar múltiples etiquetas, actuando como Segment Identifiers (SIDs), que también lo hace IPv6 a través del Segment Routing Header (SRH), llevando múltiples segmentos también. El data plane de MPLS coloca el siguiente segmento representado por la etiqueta lo más cercano al header de MAC, y retira esta etiqueta después que es usada para determinar el siguiente salto, así siempre la etiqueta actual y la siguiente estarán en el mismo lugar en relación al header MAC, que se muestra en la Figura 24.

[5] *IETF. – Internet Engineering Task Force https://www.ietf.org/*

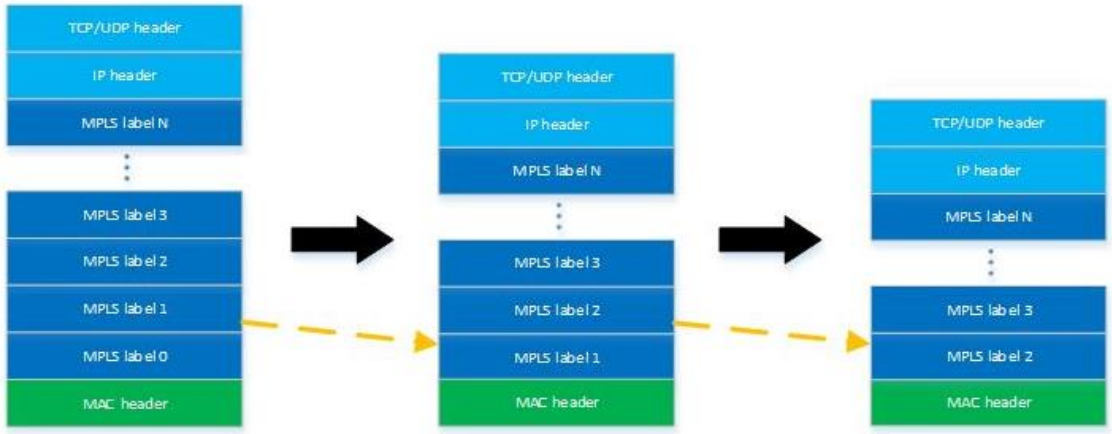


Figura 24. Stack de etiquetas MPLS
Obtenido de (Gafni, 2018)

Por el otro lado SRv6 (Segment Routing para IPv6), lleva el actual segmento como una dirección IPv6 de destino, con el resto de segmentos en el SRH, siendo el último segmento el más cercano al header MAC, y el siguiente segmento en el stack está lo más alejado del header MAC; ya que éstas ubicaciones varían se utiliza un puntero que indica la ubicación del siguiente segmento, de tal manera que ningún segmento se elimina del stack, lo cual significa una gran diferencia en velocidad en lo referente a data plane. (Gafni, 2018)

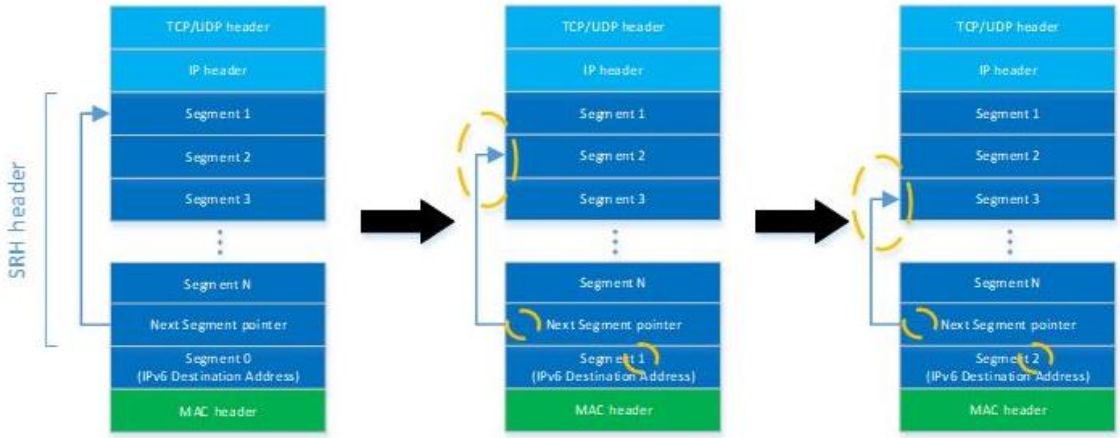


Figura 25. Stack de Segment Routing
Obtenido de (Gafni, 2018)

2.5 Características de arquitectura de SP actuales

A continuación, se presentan las topologías en lo referente a arquitectura de red entre las tecnologías MPLS y Segment Routing. El concepto a mayor detalle de Segment Routing se expone en el capítulo 3. Cabe indicar que lo referente a MPLS está expuesto en el numeral 2.4.2, por lo cual puede ser tomado como referencia para su comparación con SR.

2.5.1 Arquitectura de MPLS vs Segment Routing

La emulación de SR MPLS sirve para demostrar la factibilidad de desplegar esta tecnología sobre una infraestructura basada en MPLS y posterior SRv6 sobre un nuevo data plane de IPv6. Segment Routing introduce el paradigma de enrutamiento en el origen de acuerdo a una política compuesta por un conjunto de instrucciones llamados segmentos, donde cada segmento puede ser una instrucción basada en servicio o topológica. Los principales componentes de la arquitectura SR son:

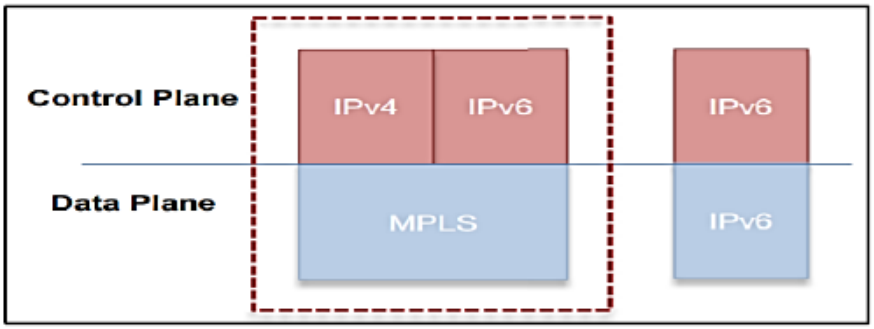


Figura 26. Control & Data Plane en un dispositivo de Service Provider
Obtenido de (Salazar Ch., Naranjo, & Marrone, 2018)

- **SR Data Plane,** define la encapsulación y el proceso de codificación del SRH en el paquete IP. Un SRH contiene una lista ordenada de segmentos identificados por su SID que puede ser global o local. Un SID global es único y puede ser anunciado a través del IGP, mientras que un SID local o SID adyacente permite mayor

granularidad de TE y es análogo a las etiquetas LDP que se asignan en entornos MPLS. Se definen cuatro tipos de segmentos que son:

- **Node-SID,** es único por cada nodo en el dominio SR.
- **Adjacency-SID,** es el identificador para un enlace de salida hacia un nodo adyacente.
- **Service-SID,** es un identificador para un servicio específico, es local.
- **Anycast-SID,** habilita la característica ECMP (*Equal-Cost Multipathing*).

Al igual que en MPLS, uno nodo en el dominio SR, ejecuta las siguientes acciones en data-plane:

- **Push,** actualiza la lista de segmentos con un nuevo segmento que pasa a ser activo.
- **Continue,** envía el paquete sin ningún cambio a la lista de segmentos
- **Next,** marca el siguiente segmento como segmento activo.

En resumen, en la siguiente tabla se muestra la comparación entre la operación de MPLS y SR:

Tabla 3. *Operaciones Segment Routing vs MPLS*

Segment Routing	MPLS
SR Header	Label Stack
Active Segment	Topmost label
PUSH	Label PUSH
NEXT	Label POP
CONTINUE	Label SWAP

Fuente: (Salazar Ch., Naranjo, & Marrone, 2018)

- **SR Control Plane,** define cómo los nodos en el dominio de SR deberían ser identificados, y cómo los nodos deberían procesar los segmentos. El Node-SID y Adjacency-SID son anunciados por el IGP que puede ser IS-IS, OSPF y vía externa como BGP. El propósito de este plano es indicar al nodo de ingreso cómo debe seleccionar la ruta a través de la red que puede ser de tres métodos:
    - **Static Configuration,** se configura un túnel SR estático, es una medida temporal.
    - **Distributed Constrained SPF Calculation,** el router de ingreso calcula la mejor ruta hacia el destino basado en criterios como clasificación de QoS, marcaje o tagging, IGP-criteria, entre otros.
    - **Uso de SDN Controller,** SR permite el uso de controlador SDN como OpenDayLight con Path Computation Element Protocol(*PCEP*) (Salazar Ch., Naranjo, & Marrone, 2018)

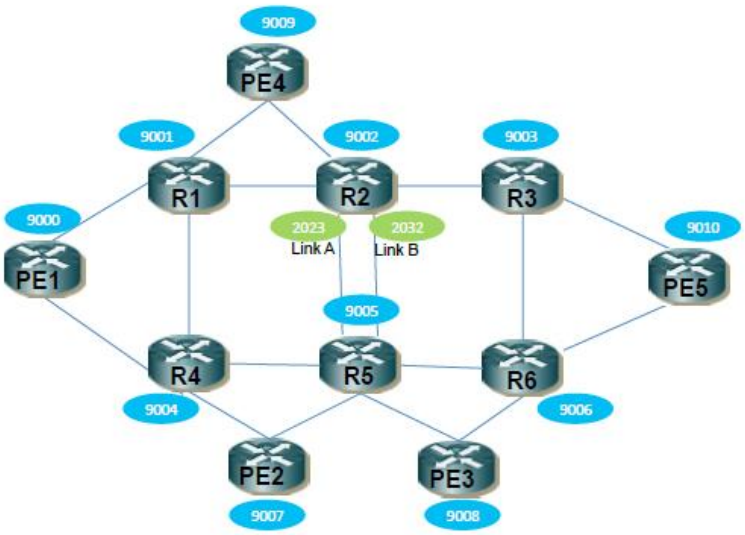


Figura 27. Topología SR-WAN
Obtenido de (Filsfils, Kumar Nainar, Pignataro, Cardona, & Francois, 2015)

De acuerdo a la Figura 27, cada nodo tiene un Node-SID, por ejemplo, R1, R5, PE3 tienen los Node-SID 9001, 9005, 9008 respectivamente; para los links de R2 se tienen los

Adjacency-SID 2023 para Link A, 2032 para Link B. PE1 puede alcanzar a PE5 usando el Node-SID 9010 en el SR Header, y el flujo que tome el paquete será balanceado sobre la ruta más corta definida por el IGP, que podría ser IS-IS, OSPF, BGP.

El router inicial puede desear que el flujo sea sobre la ruta R2-R5-R6-PE5, por lo que PE1 usará la lista de segmentos 9002, 2032, 9010. Así cuando R2 recibe el paquete, éste mueve el puntero al siguiente segmento que es 2032, realizando la operación NEXT y enviando el paquete sobre el link B. El paquete llega a R5 con segmento activo 9010, y luego de esto se ejecuta la búsqueda de la ruta más corta para alcanzar el destino PE5. Se observa entonces gran flexibilidad en la definición de la ruta que se tiene desde el router origen sin mantener estados adicionales en los routers R2 y R3, como sí, se lo hubiera requerido con RSVP-TE. (Filsfils, Kumar Nainar, Pignataro, Cardona, & Francois, 2015)

2.5.2 **Hardware y Software para Segment Routing**

En lo que respecta a sistema operativo de equipos de red, Segment Routing opera sobre IOS XR en Cisco, siendo nativo para IPv6. Otros software que soportan Segment Routing son Linux Kernel desde 4.10, FD.io VPP 17.04, P4 implementation. (Camarillo, 2019)

Equipos y soluciones que soportan esta tecnología de SRv6 en marcas reconocidas son:

- CISCO son modelos fuertes como ASR9000, NCS5500 o ASR1000 (HX, o X). (Acosta & Fonseca, 2019). Las Plataformas soportadas son IOS-XR((ASR9000, CRS-1/CRS-3, NCS5000, NCS5500, NCS6000), IOS-XE(ASR1000, CSR1000v, ASR903, ASR907, ASR920, ISR4400) (Jaksic, 2018)

Figura 28. Hardware CISCO que soporta SRv6
Obtenido de (Camarillo, 2019)

- JUNIPER junto con JunOs presenta su producto NorthStar Controller, que gracias a SDN, brinda un controlador para optimización de tráfico, automatizando la creación de rutas de ingeniería de tráfico a lo largo de la red incrementando la utilización de red y permitiendo una experiencia de red personalizada y programada. (Juniper Networks, 2020)

3 CAPTÍTULO 3 - MARCO TEÓRICO

3.1 Marco Referencial

La penetración y adopción de IPv6 a nivel mundial ha pasado desde un 5% en el año 2015 a cerca del 33% en el año 2021, lo que indica que este estándar está siendo adoptado y aceptado como un estándar nativo, por tal razón la proyección a siguientes años continuará con crecimiento exponencial. (Google, 2021)

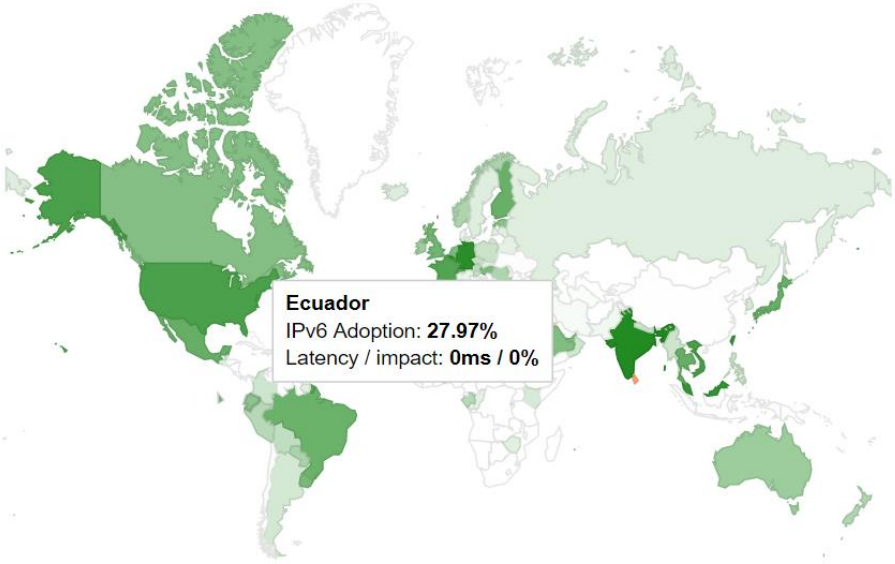


Figura 29. Adopción IPv6 por países
Obtenido de (Google, 2021)

En la Fig.29 se aprecia como la adopción de IPv6 tiene mayor fuerza en Norteamérica, Europa, India, Japón, seguido de países Australia, Brasil, Canadá, Arabia Saudita, Ecuador. En nuestro entorno se observa que el Ecuador acepta también este cambio con un 27.97% de uso del estándar IPv6, lo que debe tomarse en cuenta por parte de proveedores de servicio con miras a que sus redes sean capaces de soportar el gran crecimiento de tráfico que se va a experimentar en pocos años, así como contar con una tecnología de transporte WAN que permita operar de forma rápida y sencilla en las redes IP de nueva generación, siendo Segment Routing IPv6 el llamado a suplir estas necesidades, que para los operadores de red

va a significar gran convergencia de servicios con agilidad y alta disponibilidad. (Cai, Wielosz, & Wei, 2014)

3.2 Marco Teórico

La propuesta de Segment Routing IPv6 para Service Providers debe tomarse en cuenta, ya que por el ritmo de crecimiento de tráfico de internet y datos que se tiene, se requiere una solución en las arquitecturas WAN de los operadores que permita incrementar la escalabilidad y agilidad de red simplificando su administración, en coexistencia con los protocolos actuales y protección total de tráfico (Jaksic, 2018)

Por lo mencionado en el párrafo anterior en la actualidad los operadores de red mantienen sus operaciones basados en la red IP/MPLS, pero, por los datos a tomar en cuenta en adopción a un mundo cada vez más IPv6 y tráfico exponencial, se hace imprescindible el pensar en otra solución que supla las necesidades que la tecnología actual no brinda. Realizar un cambio drástico no siempre es la mejor solución ya que representa gastos operativos, económicos, tiempo, con el hecho de que no se tiene todo el conocimiento desarrollado y se pueden cometer errores por ignorancia. Así el hecho de poder emular una solución que presente resultados aceptables es una buena opción que servirá de referencia y garantía para que los Service Providers puedan migrar hacia redes capaces de soportar futuras demandas.

3.3 Marco Conceptual

El presente plan contempla la descripción de la situación actual en materia de redes IP que utilizan los service providers para brindar acceso a Internet y datos, exponiendo ventajas, desventajas y desafíos a los que deben estar preparados para enfrentar la demanda de servicio del mercado digital. Se realiza un estudio de Segment Routing IPv6 con la finalidad de mostrar sus ventajas frente a la tecnología actual de transporte MPLS, para luego emular y probar una red de producción de un service provider que interconecte dos sucursales para

obtener resultados que demuestren sus fortalezas a fin de determinar cuán factible es su futura implementación en redes reales.

3.4 Estudio de Segment Routing

Segment Routing SR llega a ser la solución para las demandas actuales y futuras que requiere la evolución de las WAN, debido al hecho de la transformación de negocio que requiere nuevas arquitecturas de red y que demanda de los operadores de red, que sus infraestructuras tiendan a SDN-ready(*Software Defined Network*). SR mantiene las características de una infraestructura bien diseñada y obtiene lo mejor de los conceptos de MPLS-LDP y Source Routing, brindando un control total sobre la red, mayor escalabilidad y mejor uso de la infraestructura instalada. La tendencia en TI tiende hacia SDN y Virtualización, por lo que SR emerge como la solución en WAN para manejo de flujos de tráfico y administración efectiva de tráfico *end-to-end* a lo largo de la red del proveedor de servicio, permitiendo la entrega de aplicaciones con simplicidad y escalabilidad.

No es una nueva tecnología, más se considera una variación de la técnica de *source routing*, donde el router de origen selecciona la mejor ruta de una forma predefinida y que va contenida en el mismo paquete, en lugar de estar basado en el destino y usando protocolos de enrutamiento dinámico junto a la menor métrica.

La idea de Segment Routing fue propuesta por Cisco System en noviembre de 2012 a cargo de Clarence Filsfils y el IETF que formaron en Octubre de 2013 el grupo de trabajo SPRING(*Source Packet Routing in Networking*) para el desarrollo de esta tecnología. La RFC[6] 8402, conocida como Arquitectura Segment Routing fue desarrollada en Julio de 2018.

[6] *RFC. – Request for Comments, documento técnico acerca de Internet, que incluye especificaciones y políticas producidas por 4 grupos: Internet Engineering Task Force(IETF), Internet Research Task Force(IRTF), Internet Architecture Board(IAB) e Independent Submission, https://www.rfc-editor.org/*

3.4.1 **SRv4 o SR MPLS**

SR como tecnología WAN de tunneling, satisface los nuevos requerimientos de red, al permitir alta escalabilidad y fácil solución para Traffic Engineering TE. Introduce el paradigma de source-routing de acuerdo a una política de SR compuesta por un conjunto de instrucciones llamados segmentos, donde un segmento puede ser una instrucción topológica o de servicio, que se refiere a través de su SID (*Segment Identifier*). SR puede ser aplicado en el top de la red MPLS sin cambiar su data plane. (Salazar Ch., Naranjo, & Marrone, 2018)

La arquitectura de Segment Routing está expuesta en el numeral 2.5.1, donde se presenta los dos componentes principales que son *data plane* y *control plane*, así como los tipos de segmentos que se definen.

Características relevantes de Segment Routing son:

- No usa LDP, en su lugar usa una suite de protocolos más ligera, menos adyacencias y estados de mantener.
- No tiene sincronización de IGP a LDP, por lo que se elimina delays en activar una ruta.
- Topología independiente fast reroute, usando una ruta de backup post convergencia, con protección de 50ms, sin microloops, fácil troubleshooting (Jaksic, 2018)

Segment Routing mantiene la siguiente idea: *la ruta explícita es un conjunto ordenado de instrucciones puestos dentro del paquete*, con los routers ejecutando estas instrucciones en el envío del paquete; cada instrucción se llama *segmento* y tiene su propio número llamado Segment ID (SID). La forma de codificar las instrucciones en un paquete dentro de una red MPLS es mediante el label stack, donde cada etiqueta representa un segmento, así los valores de etiquetas MPLS llevarán los valores de los SID.

Segment Routing se construye sobre el top del paradigma de envío de MPLS, y no hace cambios en la forma en que los paquetes etiquetados son enviados. Las políticas de MPLS de control plane que se aplican son:

- Para ciertos tipos de segmentos, las etiquetas tienen preferiblemente valores idénticos en todos los routers del domino SR y por tanto tienen significado global
- Las etiquetas para los segmentos son anunciadas por OSPF, IS-IS. LDP no se usa.

3.4.2 Clases de Segmentos

En Segment Routing hay dos clases principales de segmentos:

- **Global Segment:** es un valor de SID con significado en todo el dominio SR, de tal manera que cada nodo conoce este valor y asigna la misma acción para la instrucción asociada en su LFIB. El rango para este propósito es <16000-23999>. Se lo conoce como Segment Routing Global Block (SRGB) y su rango es específico para cada fabricante.
- **Local Segment:** es un SID que tiene significado local, y sólo el router origen puede ejecutar la instrucción asociada. Los valores no están en el rango SRGB, pero si en el rango de etiquetas localmente configuradas. (The Cisco Learning Network, 2018)

3.4.3 SRv6

Es la versión de IPv6 para Segment Routing que se entiende como una lista ordenada de segmentos representada como una extensión de la cabecera de enrutamiento que se define de forma amplia en la RFC8402, donde:

- 1 segmento = 1 dirección
- 1 lista de segmento = 1 lista de direcciones en el SRH (Jaksic, 2018)

En forma breve, la estructura de IPv6 y las extensiones de su cabecera se describen a continuación para mejor comprensión del tema que se trata en este caso de estudio.

3.4.4 **IPv6**

Es la versión más nueva del Internet Protocol diseñada para reemplazar a IPv4, con los siguientes cambios (Salazar, Direccionamiento IPv6 - Bases y Fundamentos, 2016):

- **Capacidad de direcciones expandida**: IPv6 incrementa la longitud de una dirección de 32 bits a 128 bits en formato hexadecimal separados en hextetos, para soportar más niveles de direccionamiento jerárquico, más número de direcciones para nodos y simple autoconfiguración. Se define un nuevo tipo de dirección llamado *anycast* que se envía para enviar un paquete de uno a un grupo de nodos.
- **Simplificación de formato de cabecera:** Algunos campos de IPv4 se han suprimido o han quedado opcionales, para reducir el procesamiento de manejar un paquete y para limitar el costo de ancho de banda de la cabecera IPv6.

- **Soporte mejorado para opciones y extensiones:** Cambios en la cabecera permiten envío más eficiente, límites menos estrictos en la longitud de las opciones permitiendo nuevas opciones a futuro.

- **Capacidad de etiquetado de flujos:** Se añade una nueva capacidad para habilitar el etiquetado de paquetes que pertenecen a un tráfico en particular, para lo cual el nodo que hace el envío solicita un especial manejo, como QoS o real time.

- **Capacidad de privacidad y autenticación:** Extensiones para soportar autenticación, integridad y confidencialidad de datos se especifican para IPv6.

3.4.5 **Formato de cabecera IPv6**

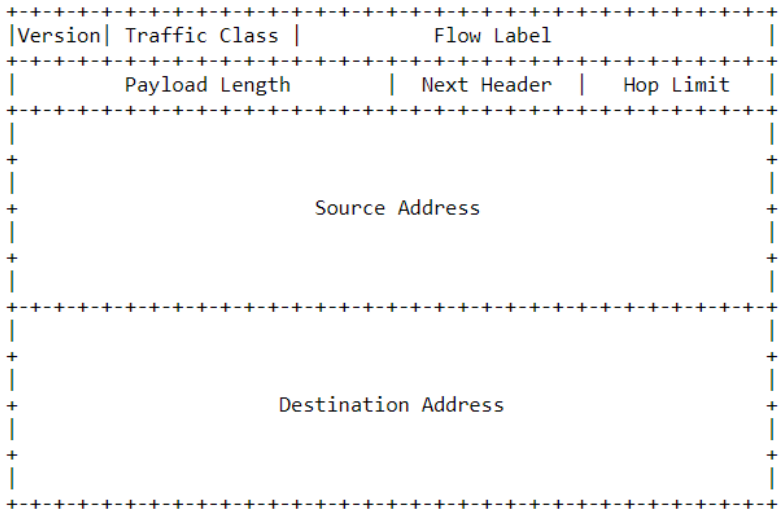


Figura 30. Formato de cabecera IPv6
Obtenido de (Deering & Hinden, 1998)

- **Versión:** 4 bit, número de versión de Internet Protocol
- **Traffic Class:** 8 bit, campo de traffic class
- **Flow Label:** 20 bit, etiqueta de flujo
- **Payload Lenght:** 16 bit, longitud del payload de IPv6, cualquier extensión de la cabecera se considera parte del payload
- **Next Header:** 8 bit selector, identifica el tipo de cabecera.
- **Hop Limit:** 8 bit, se decrementa en pasos de 1, por cada nodo que envía el paquete, si este valor llega a cero, el paquete se descarta.
- **Source Address:** 128 bit, dirección de quien origina el paquete.
- **Destination Address:** 128 bit, dirección del destinatario.

3.4.6 **Extensión de encabezados IPv6**

En IPv6, información opcional de la capa de internet es codificada en cabeceras separadas que pueden ser colocadas entre la cabecera IPv6 y la cabecera de capa superior en un

paquete. Los tipos de extensiones se identifican por el valor de Next Header, así un paquete IPv6 puede llevar cero, uno o varias extensiones, como se observa en la Figura 31,

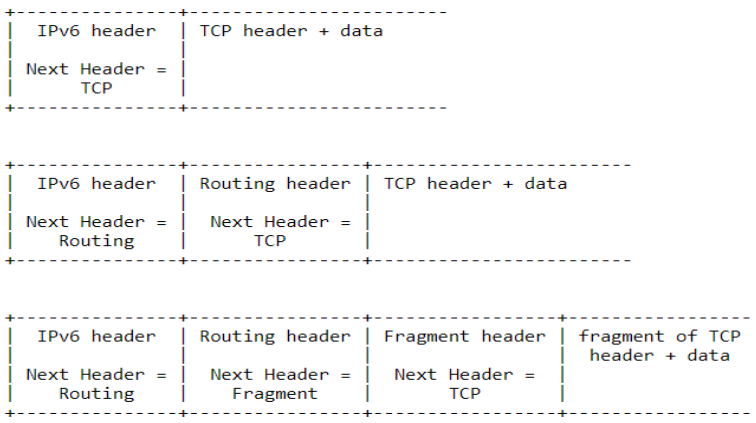


Figura 31. IPv6 Extension Headers
Obtenido de (Deering & Hinden, 1998)

Las siguientes extensiones de cabeceras se dan en una implementación completa de IPv6:

- Hop-by-Hop Options
- Routing (Type 0)
- Fragment
- Destination Options
- Authentication
- Encapsulating Security Payload (Deering & Hinden, 1998)

3.4.7 Cabecera de SRv6 SRH

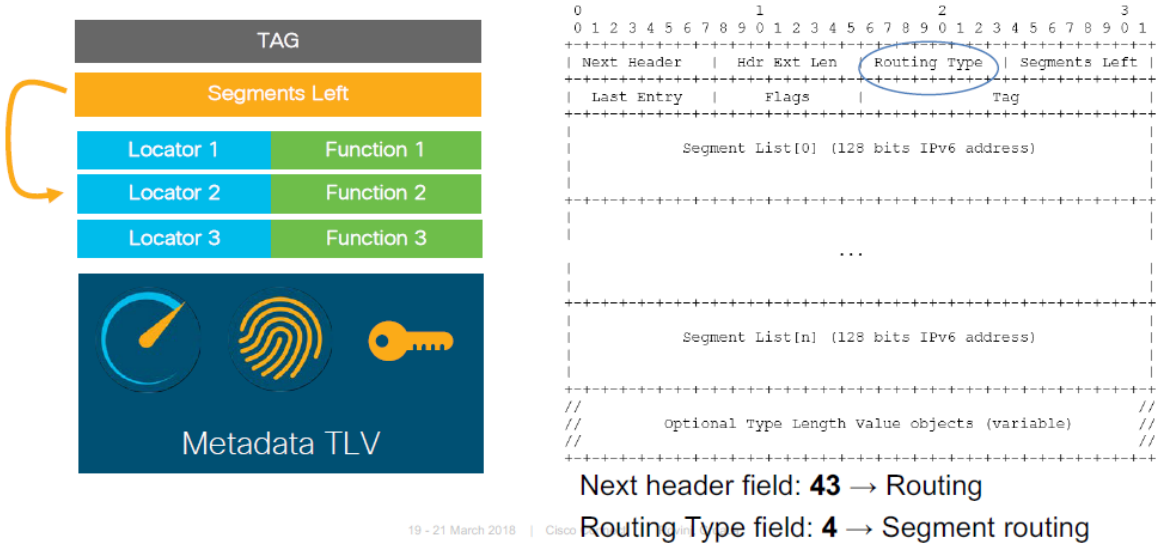


Figura 32. SRv6 Header
Obtenido de (Jaksic, 2018)

Cada segmento es una dirección IPv6, estos segmentos se codifican en reversa, el último segmento tiene índice 0, el índice del primer segmento es First Segment, el índice del segmento activo es Segment Left. El segmento activo se copia en el campo Destination Address de la cabecera IP, como se observa en la Figura 33.

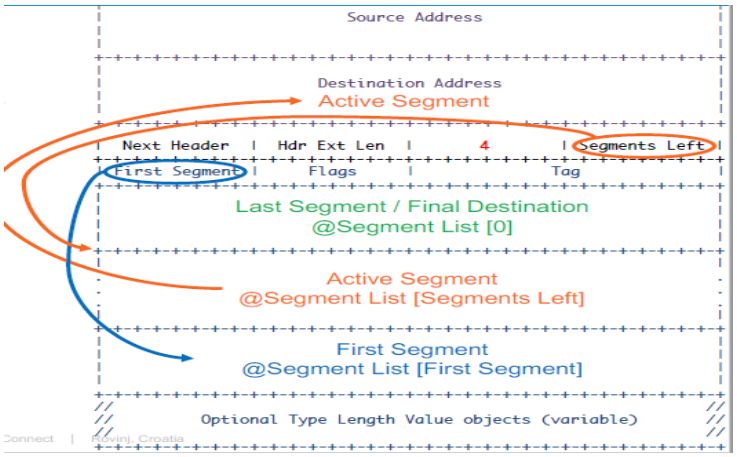


Figura 33. Codificación de Segmentos en SRv6
Obtenido de (Jaksic, 2018)

Los campos de la cabecera SRH se explican a continuación:

- **Next Header:** es el tipo de cabecera que sigue al SRH

- **Hdr Ext Len:** es la longitud de SRH en unidades de octetos
- **Segments Left:** Indica el número de segmentos faltantes, o de nodos por cruzar antes de alcanzar el destino final
- **First Segment:** Contiene el índice del último segmento de la lista
- **Flags:** Tiene 8 bits de bandera
- **Tag:** Etiqueta usada para paquetes de clase o grupo de paquetes que comparten las mismas propiedades
- **Segment List:** direcciones IPv6 de 128 bits que indica el *n* segmento de la segment list. Esta lista inicia con el último segmento de la SR policy o ruta; es decir el primer elemento de la lista hace referencia al último segmento de la ruta, el segundo elemento de la lista hace referencia al penúltimo segmento de la ruta y así hasta llegar el primer segmento de la ruta. (CISCO SYSTEMS, 2019)

Como ejemplo a continuación se muestra la forma en que se codifica la cabecera SRH, para envío de un paquete de un origen a un destino.

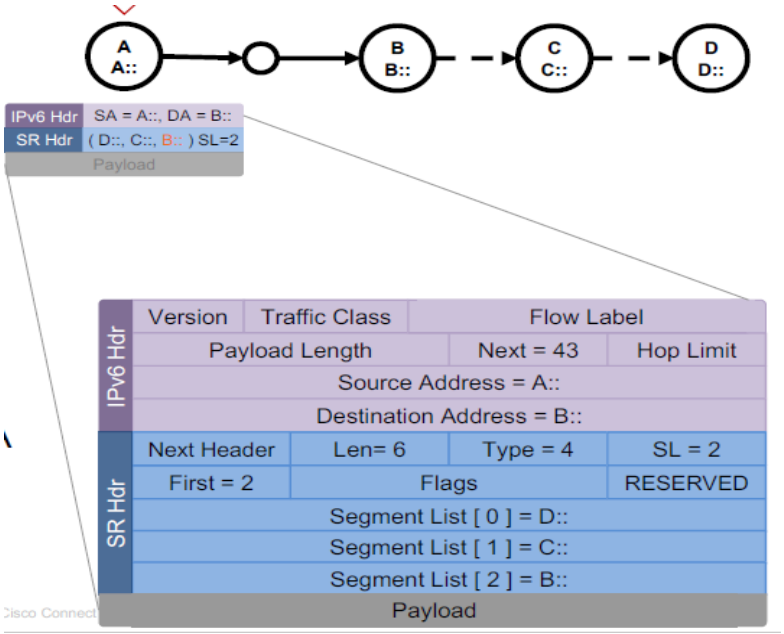


Figura 34. Codificación de SRH en IPv6
Obtenido de (Jaksic, 2018)

3.5 Ventajas y desventajas de SR frente MPLS

Como se menciona en la introducción del numeral anterior 3.4, Segment Routing resalta lo mejor de MPLS-LDP y Source Routing, con la idea de ser una arquitectura IP/MPLS diseñada con proyección a redes SDN, permitiendo el correcto balance entre inteligencia distribuida y programación-optimización centralizada con SR-TE, y amplias aplicaciones como SP, OTT/Web, GET sobre WAN, Metro/Agg, DC; MPLS e IPv6 dataplanes, SDN controller.

Segment Routing se construye sobre el top de MPLS, por lo que se puede hablar del concepto de SR MPLS, que es una lista ordenada de segmentos representados como un stack de etiquetas; de esta manera SR reúsa el dataplane de MPLS sin ningún cambio (push, swap, pop).

Dos instancias de data plane para Segment Routing se deben mencionar:

- **MPLS:** mantiene el hardware original de MPLS, y sólo se actualiza el software.

  1 segmento = 1 etiqueta

  una lista de segmentos = un stack de etiquetas

- **IPv6:** mediante la provisión de RFC2460 para extensión de cabeceras de enrutamiento de origen.

  1 segmento = 1 dirección

  una lista de segmentos = una lista de direcciones en el SRH

3.5.1 MPLS Control & Forwarding Operation con Segment Routing

En la siguiente figura se aprecia que dentro del entorno de MPLS para poder ejecutar Segment Routing, no hay cambios en control o forwarding plane, por lo que la distribución de etiquetas IGP o BGP para IPv4, IPv6 en forwarding plane es lo mismo que en MPLS.

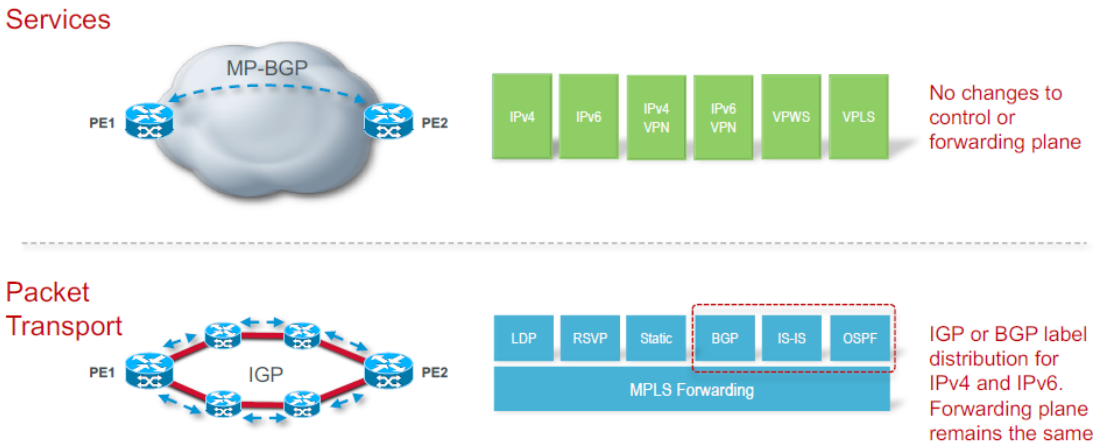


Figura 35. Servicios y operaciones de SR sobre MPLS
Obtenido de (Jaksic, 2018)

La codificación del SID se da para:

- **Prefix SID:** La etiqueta se asigna del SRGB, que se anuncia dentro del IGP vía TLV.
- **Adjacency SID:** Tiene significado local, y se aloja automáticamente por el IGP para cada adyacencia.

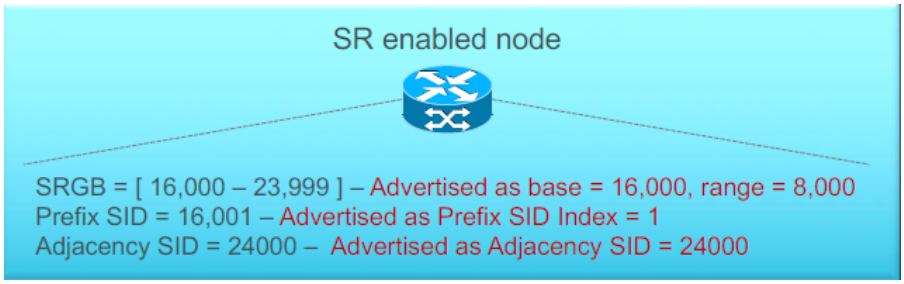


Figura 36. Codificación de SID
Obtenido de (Jaksic, 2018)

3.5.2 **Coexistencia con otros MPLS LDP**

La arquitectura MPLS permite la interacción de múltiples protocolos de distribución de etiquetas como LDP, RSVP-TE y SR control plane (Salazar Ch., Naranjo, & Marrone, 2018). Cada administrador de etiquetas en los nodos reserva un rango de etiquetas SRGB para control plane, asegurando que todas las etiquetas dinámicas están fuera del bloque SRGB y que son únicamente alojadas.

Cada LSR debe asegurar que puede interpretar sus etiquetas entrantes, que son:

- **Adjacency segment,** etiqueta única localmente alojada por el administrador de etiquetas.
- **Prefix segment,** el operador asegura el alojamiento único de cada etiqueta dentro del SRGB.

Hay que considerar que las etiquetas SR y LDP existen para un prefijo, por defecto LDP será preferida, pero SR puede ser configurado para ser preferido.

3.5.3 **MPLS LFIB con Segment Routing**

La tabla LFIB[7] es distribuida por el IGP (IS-IS, OSPF).

La diferencia entre SR y MPLS es que la tabla de forwarding (nodos + adyacencias) se mantiene constante a pesar del número de rutas en una topología Full-Mesh ISP, como se aprecia en las Figuras 37 y 38.

[7] *LFIB. – Label Forwarding Information Base, es la tabla que contienes las etiquetas asociadas a un prefijo destino con su respectiva interface de salida, https://borrowbits.com/2018/09/entendiendo-las-bases-de-mpls-casi-desde-cero/?cn-reloaded=1*

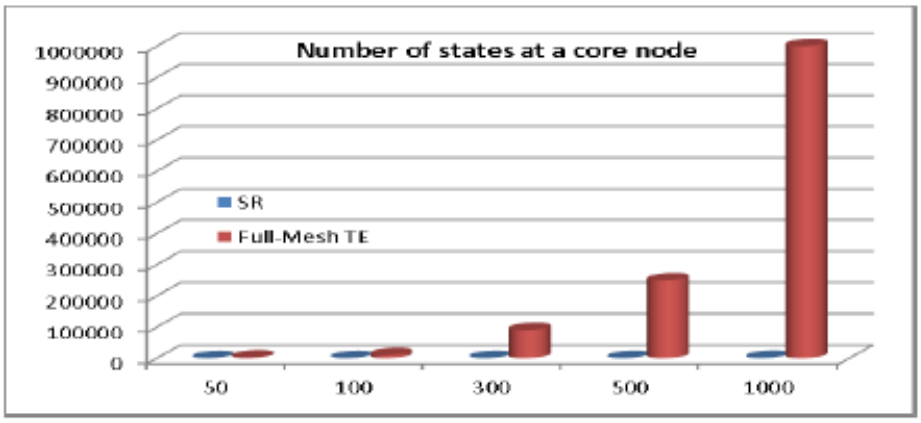


Figura 37. SR vs MPLS routing states
Obtenido de (Jaksic, 2018)

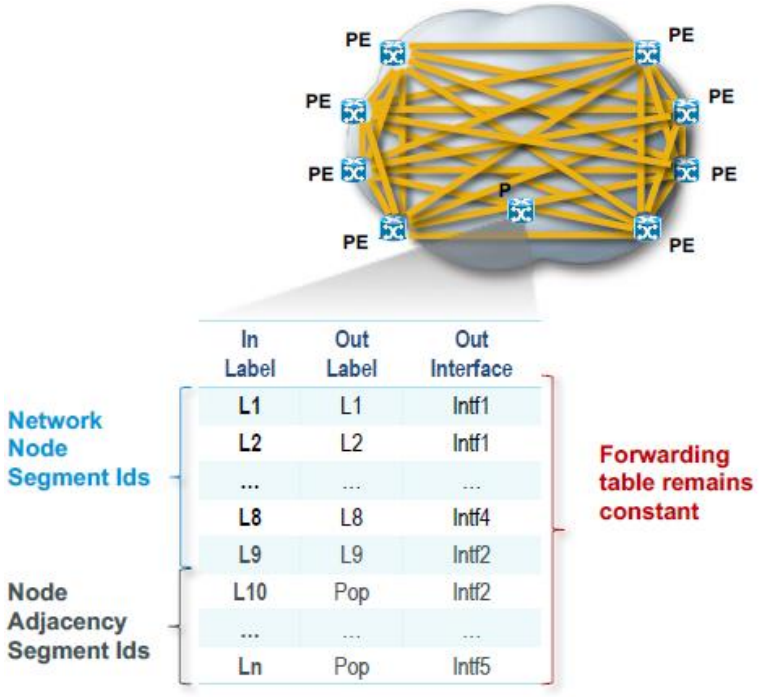


Figura 38. LFIB en Segment Routing
Obtenido de (Jaksic, 2018)

Como resumen al numeral 3.5, Segment Routing cumple con las mismas actividades que MPLS realiza y trae consigo un nuevo paradigma de codificación para el estado de envío o forwarding dentro del mismo paquete como un stack de etiquetas, dando apertura a nuevas aplicaciones. Visto desde la parte de control plane, Segment Routing confía en extensiones que se construyen desde los protocolos link-state para anunciar los SID y para proveer en

detalle la topología de red requerida para cumplir con operaciones de source routing. La reducción de complejidad operacional mientras se simplifica el proceso de envío garantiza que Segment Routing sea una tecnología atractiva a ser implementada por los proveedores de servicios en ambientes donde la simplificación marca la diferencia con las demandas que exigen los clientes.

3.6 Calidad de Servicio QoS

La Calidad de Servicio tiene distintos criterios que desde el punto de vista:

- **del usuario final** es la *percepción que tiene el usuario sobre el correcto desempeño de sus aplicaciones* en voz (llamada clara), video(alta calidad sin interrupciones), datos(bajo tiempo de respuesta)
- **del administrador de red** su objetivo es maximizar el uso de ancho de banda, sin afectar el desempeño de las aplicaciones, esto se lo puede cuantificar con parámetros como control de retardo, jitter, pérdida de paquetes.

La necesidad de habilitar QoS tiene su justificación en optimizar el ancho de banda en aplicaciones empresariales de voz, video y datos, con el fin de mejorar la productividad empresarial al clasificar y priorizar aplicaciones críticas. QoS ayuda a mantener la red disponible en eventos como ataques de DoS o gusanos, mediante su identificación y acciones correctivas.

Las operaciones básicas en QoS – DiffServ son

- **Clasificación y Marcaje:** se realiza identificación y priorización
- **Encolamiento y Descarte:** administración y eliminación
- **Operaciones post-encolamiento:** procesamiento y envío (Salazar, 2016)

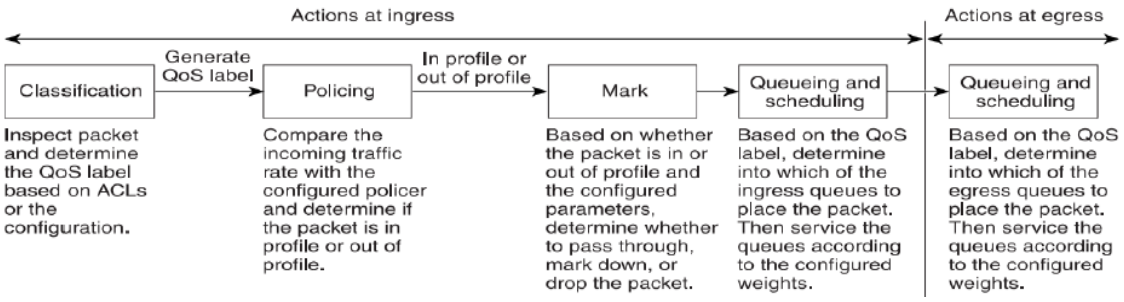


Figura 39. Modelo básico de QoS
Obtenido de (Salazar, Fundamentos de QoS -Calidad de Servicio en Capa 2 y Capa 3, 2016)

El diseño óptimo de redes con QoS sigue las normas sugeridas por CISCO que son **PPDIOO** (*Prepare, Plan, Design, Implement, Operate, Optimize*) (Salazar Ch. & Chafla, Empleo de path-control tools en una red empresarial moderna mediante políticas de enrutamiento, 2015)

3.6.1 Clases de Servicio, clasificación y marcaje

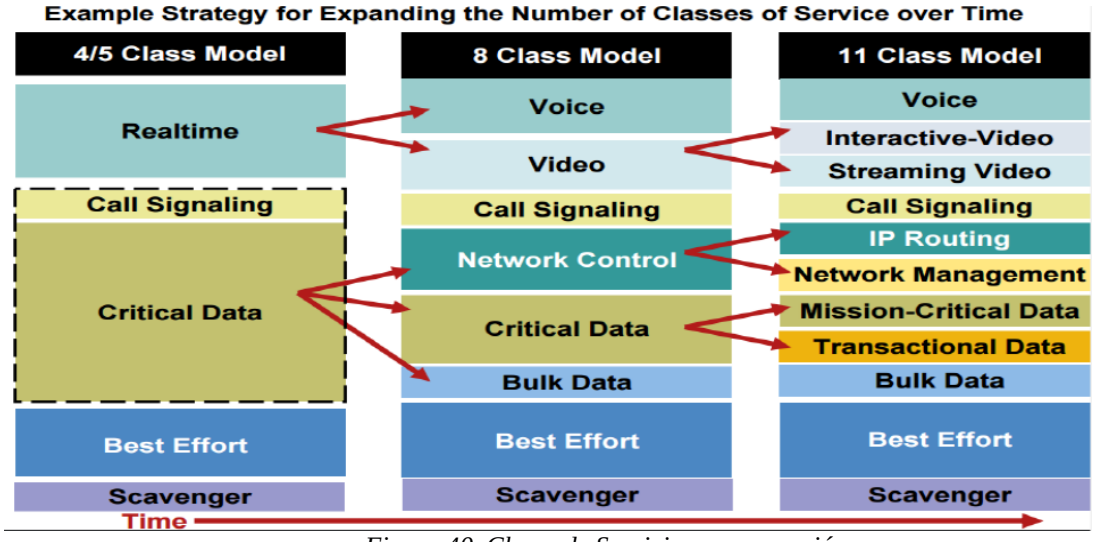


Figura 40. Clases de Servicio y su expansión
Obtenido de (Salazar, Fundamentos de QoS -Calidad de Servicio en Capa 2 y Capa 3, 2016)

- **Clasificación en Capa 2:** Toma el nombre de CoS (*Class of Service*) en capa 2 relacionado a Ethernet 802.1Q. Se observa que dentro del campo TAG de 4 bytes, 3 bits se destinan para CoS, de donde se definen 8 clases de servicios, por ejemplo un servicio de VoIP tiene CoS de 5.

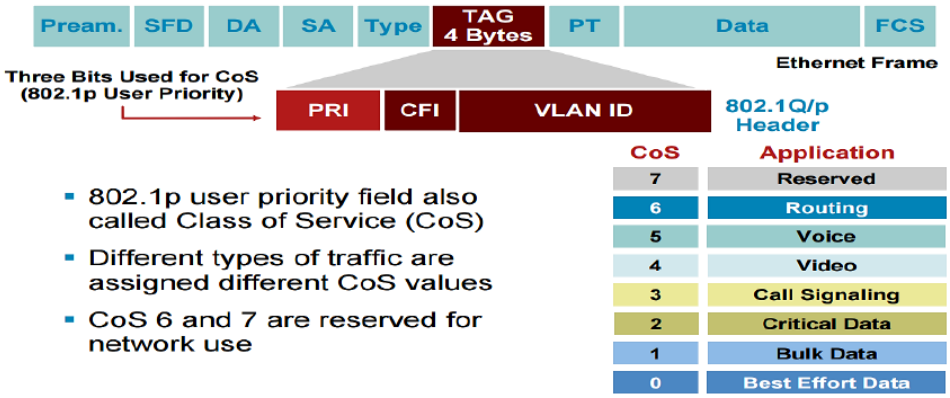


Figura 41. Trama Ethernet 802.1Q – CoS
Obtenido de (Salazar, Fundamentos de QoS -Calidad de Servicio en Capa 2 y Capa 3, 2016)

- **Clasificación en Capa 3:** Toma el nombre de ToS (*Type of Service*) en capa 3, y se definen los puntos de IP Precedence y DiffServ Code.

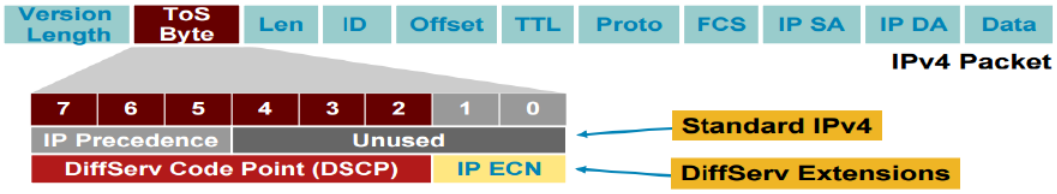


Figura 42. Paquete IP – ToS
Obtenido de (Salazar, Fundamentos de QoS -Calidad de Servicio en Capa 2 y Capa 3, 2016)

Como se ve en el campo ToS expandido se distinguen las partes:

  - **IPv4**: son los tres bits menos significantes que se conoce como IP Precedence
  - **DiffServ:** son los seis bits más significantes que se conocen como DiffServ Code Point (DSCP), dejando dos bits para control de flujo. DSCP es compatible con IP Precedence.

- **Clasificación y Marcaje para IPv6:** En IPv6 se usa el campo Traffic Class que es igual que ToS en IPv4. También se añade el campo Flow Label de 20 bits donde paquetes etiquetados pertenecen a flujos específicos, este campo puede ser usado para

solicitudes especiales del remitente, esta etiqueta no debería ser modificada por routers intermedios, estos campos se aprecian en la Figura 30.

- **NBAR:** Network Based Application Recognition es un mecanismo de clasificación de Cisco para realizar clasificación de acuerdo al protocolo

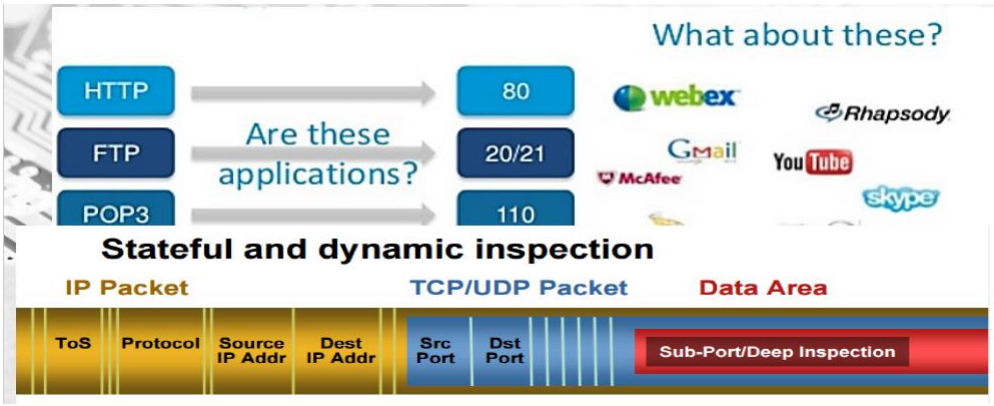


Figura 43. Clasificación por NBAR
Obtenido de (Salazar, Comunicaciones Unificadas y VoIP - PUCE, 2019)

3.6.2 Tipos de encolamiento

QoS usa técnicas de colas para clasificar el tráfico y ordenarlos de acuerdo a las clases para darles su respectiva prioridad; las técnicas de encolamiento de QoS son:

- **First-IN-First-OUT (FIFO),** no da prioridad, por tanto se considera *best effort*, ya que el primer paquete en llegar es despachado por una salida, no se recomienda para aplicaciones de voz / video.
- **Priority Queuing (PQ),** Cuatro colas de priorización (*High, Medium, Normal, Low).* Colas con más prioridad pueden consumir más recursos, *resource starvation.*
- **Customer Queuing (CQ),** Dieciséis colas configurables por el administrador, a pesar de solucionar a PQ, no da garantía de bajo delay.
- **Weighted Fair Queuing (WFQ),** Colas definidas por flujos, donde flujos con menor consumo de ancho de banda se prefieren sobre otros, no se usa para voz/video.

- **Class-Based Weighted Fair Queuing (CBWFQ),** permite hasta 256 clases de tráfico, se puede implementar con MQC (*Modular QoS CLI*) de Cisco, no garantiza retardo pero sí ancho de banda. Emplea *class-map* para asegurar AB para un tráfico específico, y usa una clase por defecto para tráfico que no ha sido designado del AB restante. Por su falta de garantía de delay no se recomienda para voz/video.
- **Low-Latency Queuing (LLQ),** es el método que se recomienda para voz/video ya que en tiempo real brinda niveles adecuados de calidad para congestiones momentáneas, es la combinación de PQ + CBWFQ, donde la voz se trata con PQ y el resto de tráfico se trata con CBWFQ.

Figura 44. Manejo de colas con LLQ
Obtenido de (Salazar, Comunicaciones Unificadas y VoIP - PUCE, 2019)

3.6.3 Técnicas de Marcaje por DSCP-PHP

Para el marcaje a través de DSCP existen algunos comportamientos *Per-Hop-Behavior:*

- **Expedited Forwarding (EF),** destinado para tráfico de voz, se procesa de inmediato para tener la menor latencia, y se debe aplicar políticas para evitar consumo de AB de tráfico de colas EF. Equivalente a DSCP 46.
- **Class Selector(CSx),** define valores DSCP y PHBs, permitiendo compatibilidad con IPP(*IP Precedence*) de valores 1-7 (CS1=DSCP8, CS2=DSCP16, CS3=DSCP24, CS4=DSCP32, …, CS7=DSCP56)
- **Assured Forwarding (AFxy),** define cuatro clases de prioridad (*mayor clase mayor prioridad*) y tres tipos de probabilidad de descarte (*menor probabilidad menos*

posibilidad de descarte). La letra X es el valor de IPP y la letra Y es *drop preference.* Existe la regla 8x+2y para transformar de AF a DSCP.

- **Best Effort (BE),** es valor de marcaje por defecto, DSCP 0. (Salazar, Fundamentos de QoS -Calidad de Servicio en Capa 2 y Capa 3, 2016)

3.6.4 QoS en Segment Routing

De acuerdo a los mecanismos de envío y procesos de control plane, es necesario garantizar AB en la ruta de segment routing para satisfacer los requerimientos de QoS del servicio. El atributo de AB garantizado puede ser aplicado a los segmentos de nodo o adyacencia, se presentan los siguientes casos

- **AB garantizado en ruta SR-TE *end-to-end,*** Con el AB garantizado de node segment y adjacency segment, SR puede calcular el AB garantizado de la ruta SR-TE end-to-end. Así este requerimiento puede ser aplicado en SR, para satisfacer los requerimientos de QoS de diferentes servicios.
- **AB garantizado en nodo frontera,** cuando el tráfico deja el dominio de red, el proceso de QoS puede ser aplicado para garantizar el servicio hacia el nodo vecino en el otro dominio de red. El proceso de QoS puede ser indicado por el segmento bandwidth en la ruta de segment routing. (Li & Wu, 2015)

3.7 Descripción de emulador de redes EVE-NG

EVE-NG(Emulated *Virtual Enviroment – Next Generation*) es un emulador de infraestructuras de red de TI, que permite realizar pruebas de conceptos, soluciones y ambientes de entrenamiento a empresas, centros de e-learning, grupos o personas, gracias a la emulación de sistemas operativos de varios fabricantes brindando amplias oportunidades a los profesionales en el mundo del networking. Para el profesional de TI brinda facilidades de:

- **Aprendizaje,** al permitir el entrenamiento propio con sistemas operativos de marcas como Cisco, Juniper, CheckPoint, PaloAlto, F5 y más.
- **Diseño,** se puede construir redes de acuerdo a los requerimientos para validar que un correcto diseño sea la solución óptima.
- **Eficiencia,** ya que se puede montar la arquitectura real en un ambiente seguro para pruebas, sin el riesgo de provocar daños en el mundo real.
- **Flexibilidad,** al poder trabajar con varias marcas y hacerlas interactuar a la disposición del diseñador. (EVE-NG, 2019)

Figura 45. Logo oficial de EVE-NG
Obtenido de (EVE-NG, 2019)

El emulador que también permite trabajar con imágenes de sistemas operativos de equipos de networking y principal competidor de EVE-NG es GNS3. Ambos se parecen en el hecho que para emular los sistemas operativos utilizan los motores llamados Qemu y Dynamips. Sus

diferencias radican en que GNS3 se presenta como una aplicación visual de Windows/Linux más fácil de manejar, en tanto que EVE-NG pertenece a Linux, una distribución de Ubuntu que contiene los paquetes y scripts necesarios para desplegar una interface web. Se reporta también que GNS3 consume mucho recurso de CPU/RAM, más esto es algo de experiencia ya que profesionales de TI que saben utilizar este emulador trabajan sin mayores problemas. EVE-NG se puede instalar como máquina virtual sobre VMWare en imagen **.OVA**, o sobre un servidor físico como **.ISO**. GNS3 permite añadir las imágenes que se desean simular, en tanto que EVE-NG sólo deja crear topologías con imágenes que se hayan instalado, es decir que para montar una arquitectura con un equipo específico, se debe subir el archivo a un directorio (/opt/unetlab/addons/dynamips/) y luego ejecutar tareas para que el sistema pueda ver este equipo, y esto se realiza por cada equipo lo cual es una tarea no tan amigable con usuario y exige destrezas sobre todo en Linux.

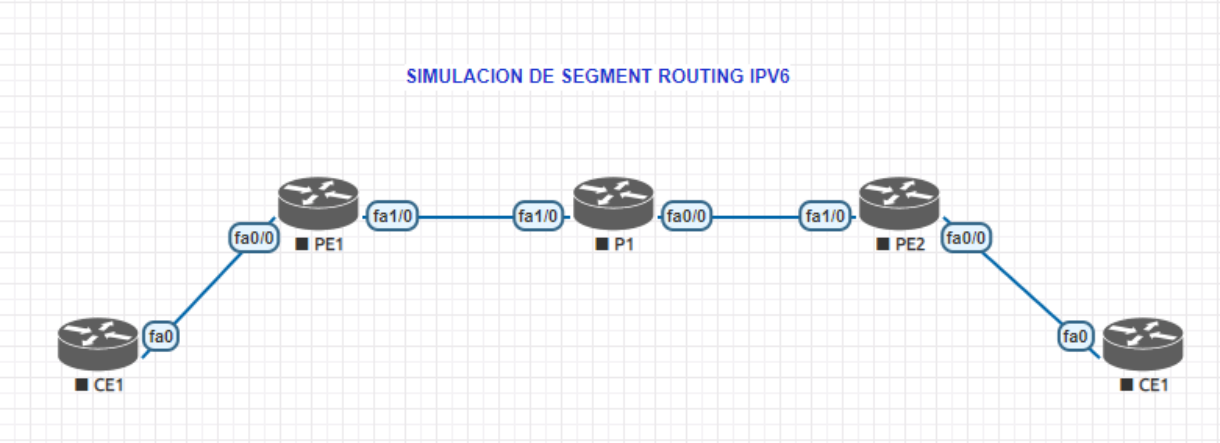


Figura 46. Arquitectura de red en EVE-NG
Fuente: Elaboración propia

Una característica y ventaja que destaca en EVE-NG, es la posibilidad de seguir aumentando capacidad de RAM/CPU/HDD/SDD a la máquina virtual, con lo cual existe una abstracción del hardware que no va a afectar al emulador que en ese momento se está ejecutando. De esta manera EVE-NG es un emulador de redes robusto, para aplicaciones que van desde lo

personal hasta lo empresarial, y escalable a medida que los requerimientos crecen. (Cabrera, 2018)

En el Anexo 1 se detalla la instalación y puesta en operación de EVE-NG, que puede ser ejecutado sobre la máquina local o también sobre cualquier cloud, que para el presente caso de estudio está montado sobre Google Cloud dado que éste sitial auspicia de forma gratuita el uso de su infraestructura para proyectos educativos por un año. Con esta solución se obtiene una máquina virtual accesible a través de una IP pública, con alta disponibilidad y que no sacrifica recursos de procesamiento local, ya que son servidores robustos que permiten elegir características de CPU, memoria y almacenamiento de acuerdo a las necesidades del cliente.

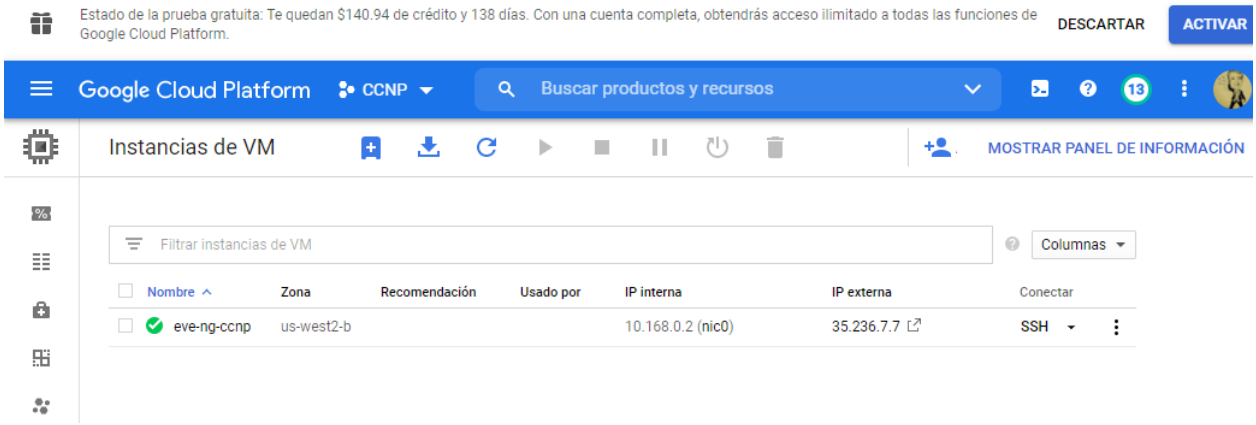


Figura 47. Entorno de Google Cloud
Fuente: Elaboración propia

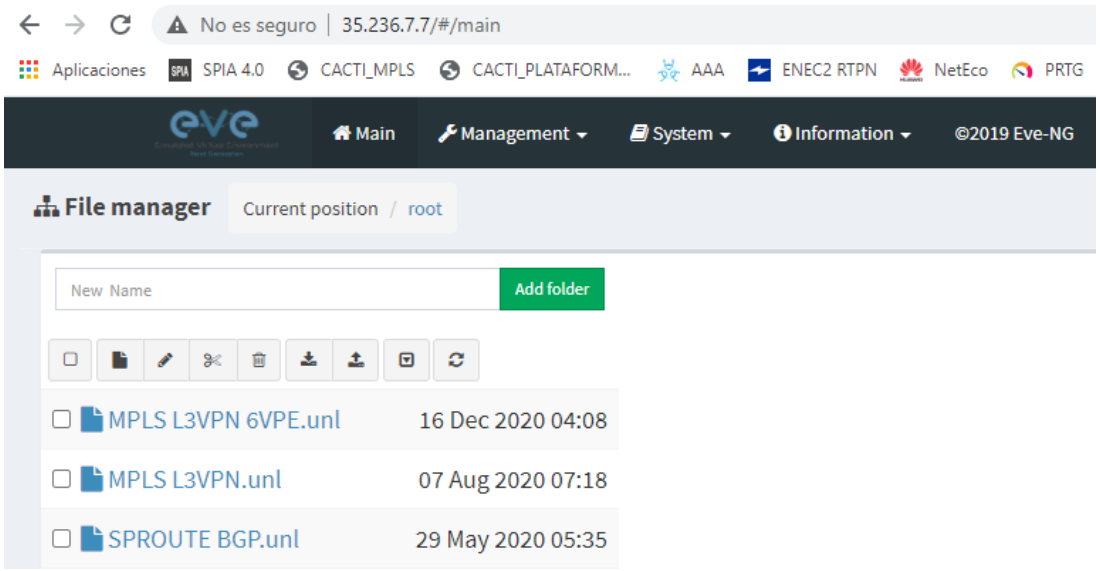


Figura 48. Pantalla principal de EVE-NG sobre Google Cloud
Fuente: Elaboración propia

4 CAPITULO 4 - SIMULACIÓN Y RESULTADOS

En este capítulo se presenta la emulación de una arquitectura de red utilizada por un SP para comunicar a dos oficinas sucursales de un cliente (Matriz-Sucursal), para lo cual se muestran tres escenarios que servirán para al final de este capítulo realizar una comparativa sobre las ventajas y desventajas de la implementación de la tecnología Segment Routing IPv6 frente a las actuales.

4.1 MPLS L3VPN

Hoy en día MPLS-L3VPN es el principal modelo desplegado por los SP para establecer comunicación entre dos puntos o más puntos remotos de un cliente, que en la práctica son varios clientes de diferentes empresas en un mismo nodo y que gracias al uso de VRF (*Virtual Routing Forwarding*) es posible cumplir este propósito administrado en el equipo de frontera del SP. Este modelo consiste en la red del cliente conformada por los equipos CE(*Customer Edge*) que se pueden conectar al proveedor de servicio mediante rutas estáticas, protocolos de enrutamiento interno como IS-IS, OSPF, EIGRP, RIP o externos como BGP; y la red del SP conformada por los equipos PE*(Provider Edge*) que se conectan con el cliente y los equipos P(*Provider*) que se pueden conectar con otros equipos P o PE; al interior de la red del SP se utiliza un IGP(*Interior Gateway Protocol*) como IS-IS para intercambiar rutas entre los equipos PE y P; y se utiliza MPLS para realizar el envío y conmutación de paquetes en base a etiquetas con el fin de obtener rapidez. Así también en este backbone se puede realizar el envío de tráfico de una o varias VPN *(Virtual Private Network*) gracias a la implementación de MP-BGP (*Multi Protocol-BGP*).

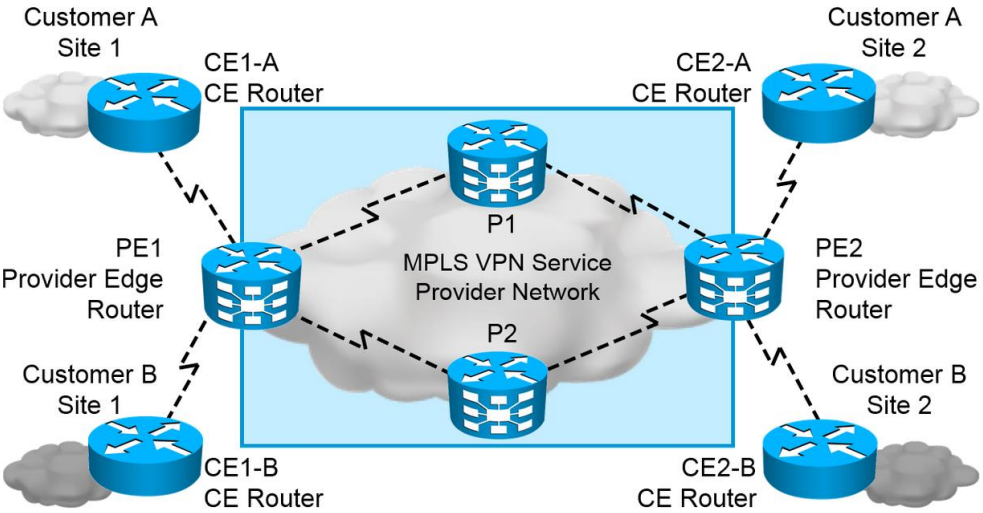


Figura 49. Componentes de MPLS VPN
Obtenido de (CISCO, 2014)

La arquitectura MPLS VPN permite a los equipos PE participar en el enrutamiento del cliente, con lo cual se mantiene un óptimo enrutamiento, a su vez que los PE pueden llevar diferentes rutas para cada cliente, con lo cual se puede usar el overlapping de direcciones. (CISCO, 2014)

El PE aísla el tráfico de varios clientes que se conectan a éste equipo, por lo cual cada cliente tiene una tabla independiente de enrutamiento VRF[8], lo cual sería como tener una conexión punto a punto, Los equipos P realizan la conmutación de etiquetas y desconocen del enrutamiento de las VPN, así como los equipos CE desconocen la existencia de los equipos P, siendo la red del SP transparente al cliente. Para gestionar muchas VPNs se debe configurar MP-BGP entre los equipos PE con el fin de llevar las rutas de varios clientes.

4.1.1 **Propagación de rutas a través del backbone**

Para llevar las rutas del cliente de un PE a otro PE a través del backbone del SP, conservando la independencia de direccionamiento, es necesario habilitar el protocolo BGP4, que en el

[8] *VRF. – Virtual Routing and Forwarding, es la tecnología que permite a un router virtualizar las tablas de enrutamiento, es decir en el mismo hardware pueden existir varias tablas de enrutamiento independientes, con lo cual se puede tener overlapping de direcciones*

entorno de MPLS VPN, con la ayuda de un prefijo de 64 bits conocido como RD (*Route Distinguisher*) convierten a un direccionamiento de 32 bits que puede estar con overlapping en una única dirección de 96 bits que puede ser transportada entre PEs. Esta dirección se la conoce como VPNv4 o VPNv6 (dependiendo de la versión del protocolo IP usado). Las sesiones que se establecen entre PEs a través de BGP se conocen como sesiones MP-BGP y permiten transportar los prefijos de un cliente entre PE a través del dominio de MPLS. Los RD se configuran por VRF en cada PE. (CISCO, 2014). La estructura de un RD es la que se muestra a continuación.

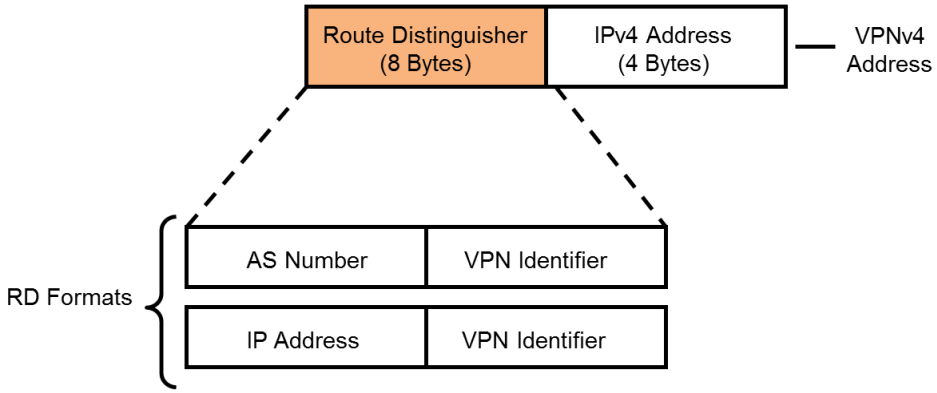


Figura 50. Estructura de Route Distinguisher
Obtenido de (CISCO, 2014)

Los RT (*Route Targets*) permiten a una VRF participar en más de una VPN, que se anuncian como comunidades extendidas de MP-BGP, para lo cual se usa Import RT / Export RT.

4.1.2 **La etiqueta VPN**

Para identificar el cliente de destino, el equipo Egress PE genera la etiqueta VPN de 4bytes a través de MP-BGP hacia el equipo Ingress PE que se usa como una etiqueta interna en el proceso de MPLS, y que se realiza una solo vez. De esta manera la etiqueta de VPN sirve para que el último equipo del backbone de MPLS sepa a donde direccionar dicho paquete de una VPN específica (CISCO, 2014)

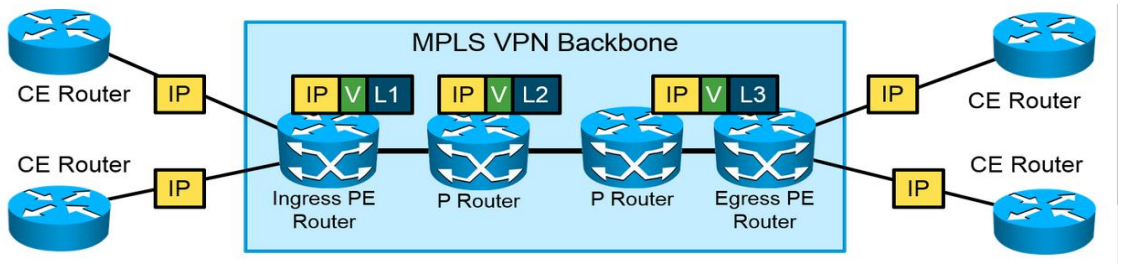


Figura 51. Esquema de alojamiento de VPN label
Obtenido de (CISCO, 2014)

Por el proceso de PHP (*Penultimate Hop Popping*) de MPLS, se extrae la etiqueta externa en el penúltimo equipo con lo cual la operación de las etiquetas de VPN en una arquitectura de MPLS es la que se muestra a continuación:

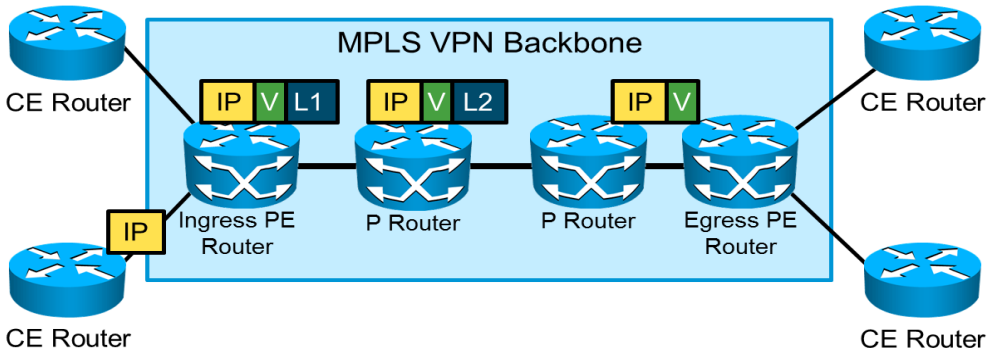


Figura 52. Operación de VPN label
Obtenido de (CISCO, 2014)

4.1.3 **Enrutamiento de MPLS L3VPN**

Con lo que se ha descrito hasta el momento se puede tener una mejor comprensión de las rutas que se intercambian entre cada equipo que participa del modelo MPLS L3VPN y que son:

- **CE:** intercambia rutas con el equipo PE a través de rutas estáticas o protocolos de enrutamiento; no tiene conocimiento del equipo P.
- **PE:** intercambia rutas de VPN con los CE a través de los protocolos de enrutamiento que se manejen por VPN, intercambia rutas de core con los equipos P y PE a través del IGP como IS-IS e intercambia rutas de VPNv4 con otros PE mediante sesiones de

MP-BGP, que llevan elementos como dirección VPNv4, comunidades extendidas (route targets), VPN label y otros atributos de BGP.

- **P:** no intervienen en el proceso de MPLS VPN, solo intercambian rutas globales con los equipos PE como enlaces de core y direcciones loopbacks. (CISCO, 2014)

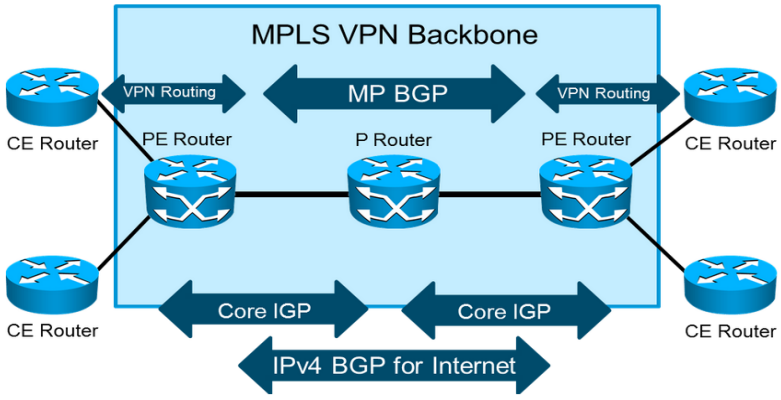


Figura 53. Modelo de enrutamiento MPLS VPN
Obtenido de (CISCO, 2014)

4.1.4 **Arquitectura MPLS L3VPN a simular**

Este caso de estudio presenta la siguiente arquitectura que se utiliza en los otros escenarios de simulación

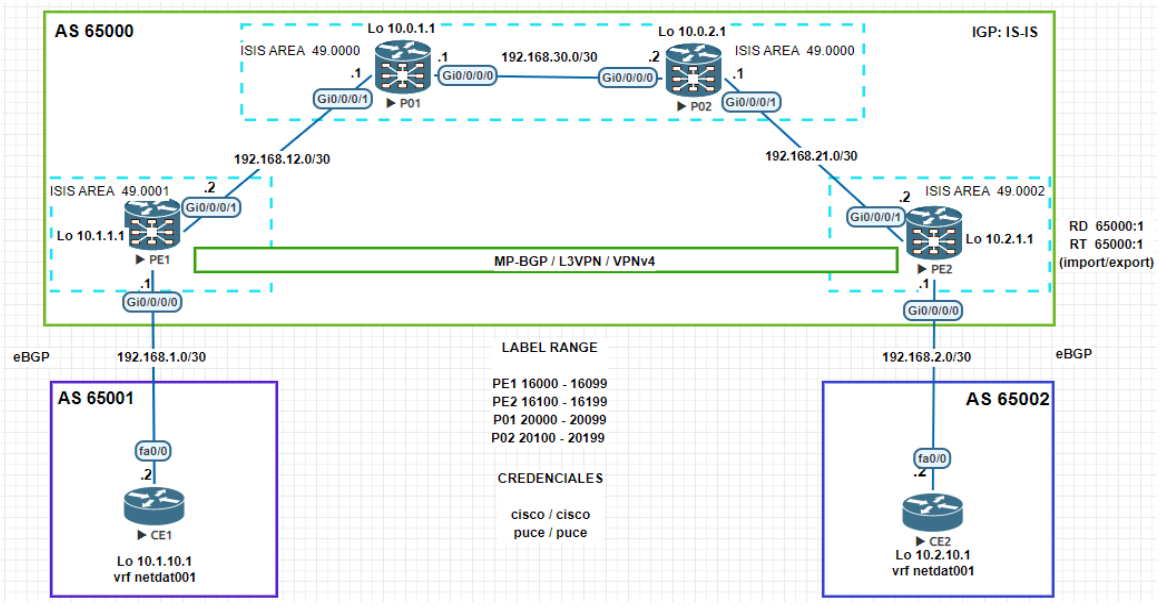


Figura 54. Arquitectura MPLS L3VPN
Fuente: Elaboración propia

Se aprecia que la arquitectura se basa en el modelo de MPLS L3VPN, con el fin de comunicar dos sucursales que se conectan al backbone del SP; cada equipo CE se conecta al SP por medio del protocolo de enrutamiento exterior de borde eBGP, para lo cual tienen asignado números de sistemas autónomos diferentes.

El backbone del SP se comporta como otro sistema autónomo que se conecta a los equipos CE a través de eBGP, en tanto que al interior establece a IS-IS de nivel 2 como protocolo de enrutamiento interior para el intercambio de rutas entre equipos PE y P con sus respectivas áreas que se distinguen por la dirección NET, esto por sus facilidades y ventajas que presenta para despliegue en ambientes de SP, y que es la base para montar MPLS con el fin de permitir el intercambio de etiquetas gracias al protocolo LDP y así obtener una conmutación rápida de paquetes. Para diferenciar el tráfico de cada cliente los equipos PE manejan el concepto de VRF, y para establecer la comunicación *end-to-end* con otro equipo PE que contenga la misma *vrf* se utiliza el protocolo MP-BGP, para poder llevar rutas de *vpnv4/vpnv6,* que se obtienen al combinar la dirección IP con el número de AS, lo que se conoce como *route distinguisher*, y finalmente para realizar la exportación/importación de estas rutas se utiliza los *route targets*. Se resalta el hecho que para desplegar la comunicación *end-to-end* en ambientes reales es mejor adoptar el concepto de route reflector, con el fin de disminuir el número de enlaces que se generan entre equipos que utilizan MP-BGP pasando de ser una topología full mesh a una topología tipo estrella ya que los equipos establecen comunicación con un solo equipo quien a su vez se encarga de propagar las rutas de vpnv4/vpnv6 hacia el respectivo destino; en este trabajo de estudio y para los escenarios de simulación no se maneja el concepto de route reflector.

En la siguiente tabla se muestra el direccionamiento utilizado

Tabla 4. *Direccionamiento de arquitectura MPLS L3VPN*

EQUIPO	INTERFACE	DIRECCIÓN	AS
CE1	Fa0/0 Lo0	192.168.1.2/30 10.1.10.1/32	65001
CE2	Fa0/0 Lo0	192.168.2.2/30 10.2.10.1/32	65002
PE1	Gi0/0/0/0 Gi0/0/0/1 Lo0 NET vrf, RD/RT	192.168.1.1/30 192.168.12.2/30 10.1.1.1/32 49.0001.0100.0100.1001.00 netdat001, 65000:1	65000
PE2	Gi0/0/0/0 Gi0/0/0/1 Lo0	192.168.2.1/30 192.168.21.2/30 10.2.1.1/32	
	NET vrf, RD/RT	49.0002.0100.0200.1001.00 netdat001, 65000:1	
P01	Gi0/0/0/0 Gi0/0/0/1	192.168.30.1/30 192.168.12.1/30	
	Lo0 NET	10.0.1.1/32 49.0000.0100.0000.1001.00	
P02	Gi0/0/0/0 Gi0/0/0/1 Lo0 NET	192.168.30.2/30 192.168.21.1/30 10.0.2.1/32 49.0000.0100.0000.2001.00	

Fuente: Elaboración propia

Con los datos presentados se configuran los equipos, cuyos scripts se detallan en anexos. Se realizan verificaciones para determinar que la arquitectura está operativa y que son las siguientes:

- **Verificación de IGP**

Esto se realiza en los equipos que pertenecen al backbone del SP con el comando *show route isis*, con lo cual se aprecia que se aprenden rutas de todos los equipos que pertenecen al backbone

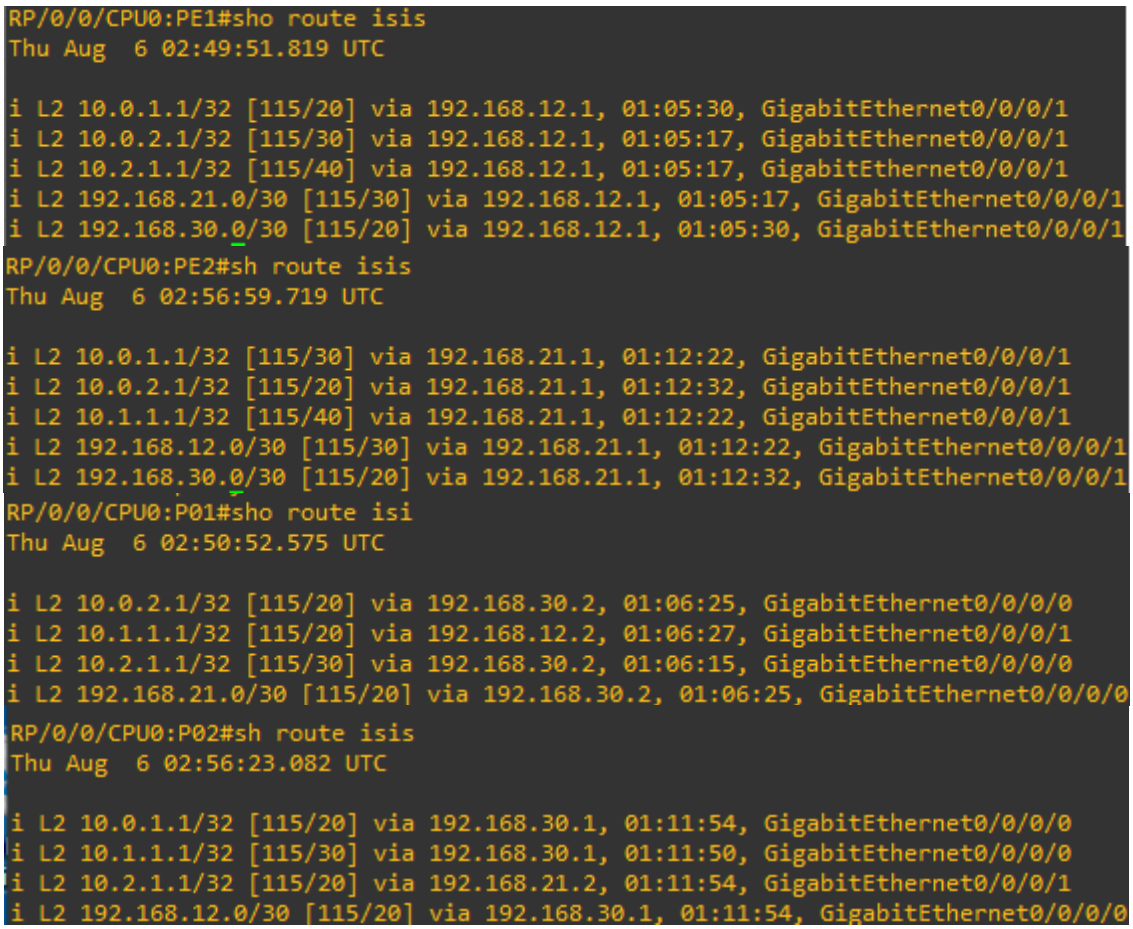


Figura 55. Verificación de IGP en MPLS L3VPN
Fuente: Elaboración propia

- **Verificación de MPLS LDP**

Una vez que se ha establecido el protocolo ISIS, se comprueba el funcionamiento de MPLS LDP, con lo cual se aprecia la asignación de etiquetas de acuerdo a lo indicado en la Figura 56, con el comando *show mpls ldp forwarding / show mpls forwarding*

```
RP/0/0/CPU0:PE1#sho mpls ldp forwarding
Thu Aug  6 03:09:25.718 UTC

Codes:
  - = GR label recovering, (!) = LFA FRR pure backup path
  {} = Label stack with multi-line output for a routing path
  G = GR, S = Stale, R = Remote LFA FRR backup

Prefix           Label   Label(s)       Outgoing     Next Hop
                 In      Out            Interface
---------------- ------- -------------- ------------ --------------
10.0.1.1/32      16000   ImpNull        Gi0/0/0/1    192.168.12.1
10.0.2.1/32      16002   20001          Gi0/0/0/1    192.168.12.1
10.2.1.1/32      16003   20003          Gi0/0/0/1    192.168.12.1
192.168.21.0/30  16004   20002          Gi0/0/0/1    192.168.12.1
192.168.30.0/30  16001   ImpNull        Gi0/0/0/1    192.168.12.1
```

```
RP/0/0/CPU0:P01#sh mpl ldp forwarding
Thu Aug  6 03:10:43.013 UTC

Codes:
  - = GR label recovering, (!) = LFA FRR pure backup path
  {} = Label stack with multi-line output for a routing path
  G = GR, S = Stale, R = Remote LFA FRR backup

Prefix           Label   Label(s)       Outgoing     Next Hop
                 In      Out            Interface
---------------- ------- -------------- ------------ -------------
10.0.2.1/32      20001   ImpNull        Gi0/0/0/0    192.168.30.2
10.1.1.1/32      20000   ImpNull        Gi0/0/0/1    192.168.12.2
10.2.1.1/32      20003   20100          Gi0/0/0/0    192.168.30.2
192.168.21.0/30  20002   ImpNull        Gi0/0/0/0    192.168.30.2
RP/0/0/CPU0:P02#sh mpls ldp forwarding
Thu Aug  6 03:12:48.174 UTC

Codes:
  - = GR label recovering, (!) = LFA FRR pure backup path
  {} = Label stack with multi-line output for a routing path
  G = GR, S = Stale, R = Remote LFA FRR backup

Prefix           Label   Label(s)       Outgoing     Next Hop
                 In      Out            Interface
---------------- ------- -------------- ------------ -------------
10.0.1.1/32      20101   ImpNull        Gi0/0/0/0    192.168.30.1
10.1.1.1/32      20103   20000          Gi0/0/0/0    192.168.30.1
10.2.1.1/32      20100   ImpNull        Gi0/0/0/1    192.168.21.2
192.168.12.0/30  20102   ImpNull        Gi0/0/0/0    192.168.30.1
```

```
RP/0/0/CPU0:PE2#show mpls ldp forwarding
Thu Aug  6 03:14:44.737 UTC

Codes:
  - = GR label recovering, (!) = LFA FRR pure backup path
  {} = Label stack with multi-line output for a routing path
  G = GR, S = Stale, R = Remote LFA FRR backup

Prefix          Label   Label(s)       Outgoing     Next Hop
                In      Out            Interface
--------------- ------- -------------- ------------ -------------
10.0.1.1/32     16102   20101          Gi0/0/0/1    192.168.21.1
10.0.2.1/32     16100   ImpNull        Gi0/0/0/1    192.168.21.1
10.1.1.1/32     16103   20103          Gi0/0/0/1    192.168.21.1
192.168.12.0/30 16104   20102          Gi0/0/0/1    192.168.21.1
192.168.30.0/30 16101   ImpNull        Gi0/0/0/1    192.168.21.1
```

Figura 56. Verificación de MPLS LDP en MPLS L3VPN
Fuente: Elaboración propia

- **Verificación de iBGP /MP-BGP / VPNv4**

En este paso se verifica que la comunicación *end-to-end* entre equipos PE para el intercambio de rutas *vpnv4* de una misma *vrf* trabaja normal. Con el comando *show bgp vpnv4 unicast summary* se observa el vecino PE con quien se va a intercambiar rutas.

```
RP/0/0/CPU0:PE1#show bgp vpnv4 unicast summary
Fri Aug  7 04:28:25.211 UTC
BGP router identifier 10.1.1.1, local AS number 65000
BGP generic scan interval 60 secs
Non-stop routing is enabled
BGP table state: Active
Table ID: 0x0   RD version: 0
BGP main routing table version 13
BGP NSR Initial initsync version 6 (Reached)
BGP NSR/ISSU Sync-Group versions 0/0
BGP scan interval 60 secs

BGP is operating in STANDALONE mode.


Process       RcvTblVer   bRIB/RIB   LabelVer  ImportVer  SendTblVer  StandbyVer
Speaker              13         13         13         13          13           0

Neighbor        Spk    AS MsgRcvd MsgSent   TblVer  InQ OutQ  Up/Down  St/PfxRcd
10.2.1.1          0 65000      64      64       13    0    0 00:59:54          2
```

```
RP/0/0/CPU0:PE2#sho bgp vpnv4 unicast summary
Fri Aug  7 04:29:09.038 UTC
BGP router identifier 10.2.1.1, local AS number 65000
BGP generic scan interval 60 secs
Non-stop routing is enabled
BGP table state: Active
Table ID: 0x0   RD version: 0
BGP main routing table version 13
BGP NSR Initial initsync version 6 (Reached)
BGP NSR/ISSU Sync-Group versions 0/0
BGP scan interval 60 secs

BGP is operating in STANDALONE mode.


Process       RcvTblVer   bRIB/RIB   LabelVer  ImportVer  SendTblVer  StandbyVer
Speaker              13         13         13         13          13           0

Neighbor        Spk    AS MsgRcvd MsgSent   TblVer  InQ OutQ  Up/Down  St/PfxRcd
10.1.1.1          0 65000      65      65       13    0    0 01:00:39          2
```

Figura 57. Verificación de MP-BGP VPNv4 en MPLS L3VPN
Fuente: Elaboración propia

- **Verificación de eBGP**

Dado que la comunicación con los equipos CE que representan las sucursales remotas del cliente se realiza mediante sesiones BGP se puede comprobar el establecimiento de sesiones eBGP entre los equipos PE – CE, a través de los comandos *show bgp vrf netdat001 summary* para los equipos PE y *show bgp summary* para equipos CE

```
RP/0/0/CPU0:PE1#show bgp vrf netdat001 summary
Fri Aug  7 05:04:00.955 UTC
BGP VRF netdat001, state: Active
BGP Route Distinguisher: 65000:1
VRF ID: 0x60000002
BGP router identifier 10.1.1.1, local AS number 65000
Non-stop routing is enabled
BGP table state: Active
Table ID: 0xe0000011   RD version: 13
BGP main routing table version 13
BGP NSR Initial initsync version 6 (Reached)
BGP NSR/ISSU Sync-Group versions 0/0

BGP is operating in STANDALONE mode.


Process       RcvTblVer   bRIB/RIB   LabelVer  ImportVer  SendTblVer  StandbyVer
Speaker              13         13         13         13          13           0

Neighbor        Spk    AS MsgRcvd MsgSent   TblVer  InQ OutQ  Up/Down  St/PfxRcd
192.168.1.2       0 65001     100     100       13    0    0 01:36:25          2
```

```
RP/0/0/CPU0:PE2#show bgp vrf netdat001 summary
Fri Aug  7 05:00:29.189 UTC
BGP VRF netdat001, state: Active
BGP Route Distinguisher: 65000:1
VRF ID: 0x60000002
BGP router identifier 10.2.1.1, local AS number 65000
Non-stop routing is enabled
BGP table state: Active
Table ID: 0xe0000011   RD version: 13
BGP main routing table version 13
BGP NSR Initial initsync version 6 (Reached)
BGP NSR/ISSU Sync-Group versions 0/0

BGP is operating in STANDALONE mode.


Process       RcvTblVer   bRIB/RIB   LabelVer  ImportVer  SendTblVer  StandbyVer
Speaker              13         13         13         13          13           0

Neighbor        Spk    AS MsgRcvd MsgSent   TblVer  InQ OutQ  Up/Down  St/PfxRcd
192.168.2.2       0 65002      96      96       13    0    0 01:32:54          2
```

```
CE1#show bgp summary
BGP router identifier 10.1.10.1, local AS number 65001
BGP table version is 5, main routing table version 5
4 network entries using 480 bytes of memory
4 path entries using 208 bytes of memory
3/2 BGP path/bestpath attribute entries using 372 bytes of memory
1 BGP AS-PATH entries using 24 bytes of memory
0 BGP route-map cache entries using 0 bytes of memory
0 BGP filter-list cache entries using 0 bytes of memory
Bitfield cache entries: current 1 (at peak 1) using 32 bytes of memory
BGP using 1116 total bytes of memory
BGP activity 4/0 prefixes, 4/0 paths, scan interval 60 secs

Neighbor        V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
192.168.1.1     4 65000     101     101        5    0    0 01:37:16        2
```

```
CE2#sho bgp summary
BGP router identifier 10.2.10.1, local AS number 65002
BGP table version is 5, main routing table version 5
4 network entries using 480 bytes of memory
4 path entries using 208 bytes of memory
3/2 BGP path/bestpath attribute entries using 372 bytes of memory
1 BGP AS-PATH entries using 24 bytes of memory
0 BGP route-map cache entries using 0 bytes of memory
0 BGP filter-list cache entries using 0 bytes of memory
Bitfield cache entries: current 1 (at peak 1) using 32 bytes of memory
BGP using 1116 total bytes of memory
BGP activity 4/0 prefixes, 4/0 paths, scan interval 60 secs

Neighbor        V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
192.168.2.1     4 65000     102     102        5    0    0 01:38:21        2
```

Figura 58. Verificación de BGP en MPLS L3VPN
Fuente: Elaboración propia

- **Verificación de rutas y prueba de conectividad**

En esta parte se verifica el intercambio de rutas entre equipos CE con lo cual se demuestra que la comunicación *end-to-end* a través del backbone del SP funciona; con el comando *show ip route* en los equipos CE se obtiene esta información

```
CE1#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

     10.0.0.0/32 is subnetted, 2 subnets
C       10.1.10.1 is directly connected, Loopback0
B       10.2.10.1 [20/0] via 192.168.1.1, 01:38:25
     192.168.1.0/30 is subnetted, 1 subnets
C       192.168.1.0 is directly connected, FastEthernet0/0
     192.168.2.0/30 is subnetted, 1 subnets
B       192.168.2.0 [20/0] via 192.168.1.1, 01:38:25
```

```
CE2#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

     10.0.0.0/32 is subnetted, 2 subnets
B       10.1.10.1 [20/0] via 192.168.2.1, 01:42:24
C       10.2.10.1 is directly connected, Loopback0
     192.168.1.0/30 is subnetted, 1 subnets
B       192.168.1.0 [20/0] via 192.168.2.1, 01:42:24
     192.168.2.0/30 is subnetted, 1 subnets
C       192.168.2.0 is directly connected, FastEthernet0/0
```

Figura 59. Verificación de rutas entre equipos CE a través de MPLS L3VPN
Fuente: Elaboración propia

También es posible realizar un seguimiento a la ruta con lo cual se puede comprobar la asignación de etiquetas que ocurre en el backbone del SP tanto para MPLS como para VPNv4, con el comando *traceroute* se tiene lo siguiente

```
CE1#traceroute 10.2.10.1

Type escape sequence to abort.
Tracing the route to 10.2.10.1

  1 192.168.1.1 28 msec 36 msec 8 msec
  2 192.168.12.1 [MPLS: Labels 20003/16105 Exp 0] 36 msec 32 msec 28 msec
  3 192.168.30.2 [MPLS: Labels 20100/16105 Exp 0] 28 msec 28 msec 12 msec
  4 192.168.21.2 [MPLS: Label 16105 Exp 0] 32 msec 20 msec 36 msec
  5 192.168.2.2 [AS 65002] 20 msec 48 msec *
```

```
CE2#traceroute 10.1.10.1

Type escape sequence to abort.
Tracing the route to 10.1.10.1

  1 192.168.2.1 28 msec 40 msec 8 msec
  2 192.168.21.1 [MPLS: Labels 20103/16005 Exp 0] 32 msec 36 msec 20 msec
  3 192.168.30.1 [MPLS: Labels 20000/16005 Exp 0] 20 msec 36 msec 16 msec
  4 192.168.12.2 [MPLS: Label 16005 Exp 0] 24 msec 44 msec 20 msec
  5 192.168.1.2 [AS 65001] 24 msec 28 msec *
```

Figura 60. Prueba de traceroute en MPLS L3VPN
Fuente: Elaboración propia

Por último se realiza una prueba de ping entre equipos CE para comprobar conectividad satisfactoria

```
CE1#ping 10.2.10.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.2.10.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 16/20/28 ms
```

```
CE2#ping 10.1.10.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.10.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 16/21/28 ms
```

Figura 61. Prueba de ping entre CE en MPLS L3VPN
Fuente: Elaboración propia

4.2 SEGMENT ROUTING PARA IPv4 o SR MPLS

Se lo conoce también como SR MPLS ya que usa el data plane de MPLS por lo que su funcionamiento es el mismo de MPLS. La analogía que se debe tener en claro y como se lo explica en el numeral 2.5.1 es que un segmento equivale a un label y una lista de segmentos equivale a un label stack, con la ventaja que la tabla LFIB se mantiene constante y no crece a medida que una ruta tenga más saltos como se lo explica en el numeral 3.5.3. Se destaca el hecho que SR no utiliza el protocolo LDP, ya que un segmento es una instrucción. Por defecto si un equipo tiene configurado MPLS y SR, se va a preferir el uso de MPLS, por lo cual se tiene que preferir el uso de SR en la configuración. Puede darse el hecho que existan nodos operando en MPLS o SR, por tanto para que estos nodos puedan interactuar es necesario utilizar un SR Mapping-server.

4.2.1 Arquitectura SR MPLS a simular

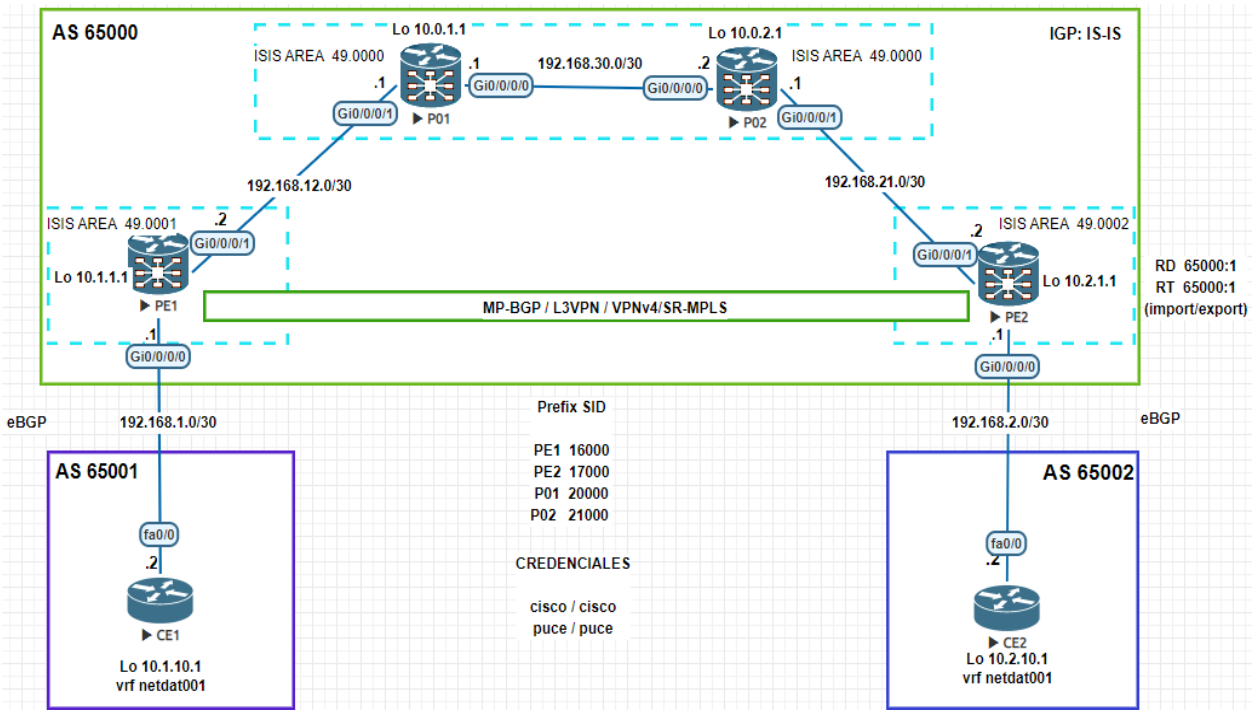


Figura 62. Arquitectura SR MPLS
Fuente: Elaboración propia

El escenario planteado es Segment Routing sobre el cloud MPLS de un SP y que es el mismo que se detalla en el numeral 4.1.4; comprende la comunicación entre dos sucursales a través del core de un SP que sigue utilizando IS-IS como IGP ya que al ser un protocolo CLNS (*Connectionless Network Service*) puede llevar cualquier protocolo de capa 3, y para ser usado en Segment Routing necesita una extensión que sirve para mantener una base de datos de los SID de nodos y adyacencias (Salazar Ch., Naranjo, & Marrone, 2018), así como *vrf* para diferenciar el tráfico de un cliente específico y el protocolo MP-BGP para establecer la comunicación *end-to-end* entre dos equipos PE para poder llevar rutas *vpnv4/vpnv6* de una misma *vrf*. La principal diferencia que se establece en el uso de SR MPLS es que no se utiliza el protocolo LDP encargado de distribuir etiquetas; en el data plane el SRH establece la lista ordenada de segmentos identificados por su SID que se anuncia a través del IGP IS-IS, lo cual es ventaja en que la tabla de forwarding se mantenga constante, a pesar del número de rutas que existan en una topología del SP, que lleva a simplificar la operación en el core de un SP, como se muestra en el numeral 3.5.3. Los scripts de los equipos se detallan en anexos. Debido a que SR MPLS, se construye sobre la arquitectura MPLS L3VPN, no se verifica la operación del IGP IS-IS, así como el intercambio de rutas de VPNv4 a través de MP-BGP, ni el funcionamiento de eBGP entre el core del SP y los clientes. A continuación, se muestran las pruebas para determinar la operación de SR MPLS:

- **Verificación de MPLS LDP desactivado**

Después de configurar SR MPLS dentro del IGP que es IS-IS, se comprueba que no se está utilizando el protocolo LDP por lo cual ya no se realiza la asignación de etiquetas; con el comando *show mpls interface gigabitEthernet **x/x/x/x** detail* se obtiene la información, además con el comando *show cef **loopback_destino*** | *include labels* se comprueba la imposición del segmento que se asigna a un equipo PE destino, desde un PE origen:

```
RP/0/0/CPU0:PE1#sho mpls int gigabitEthernet 0/0/0/1 detail
Tue Aug 18 11:29:11.349 UTC
Interface GigabitEthernet0/0/0/1:
        LDP labelling not enabled
        LSP labelling not enabled
        MPLS ISIS enabled
        MPLS enabled
```

```
RP/0/0/CPU0:PE1#sho cef 10.2.1.1 | include labels
Tue Aug 18 11:29:58.205 UTC
     local label 17000      labels imposed {17000}
RP/0/0/CPU0:PE1#
```

Figura 63. Verificación de no uso de LDP en SR MPLS e imposición de segmento
Fuente: Elaboración propia

- **Verificación de rutas y pruebas de conectividad en SR MPLS**

Se comprueba que se tiene comunicación *end-to-end* entre los equipos CE a través del core del SP que utiliza SR MPLS, con el comando *show ip route:*

```
CE1#sho ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

     10.0.0.0/32 is subnetted, 2 subnets
C       10.1.10.1 is directly connected, Loopback0
B       10.2.10.1 [20/0] via 192.168.1.1, 01:44:36
     192.168.1.0/30 is subnetted, 1 subnets
C       192.168.1.0 is directly connected, FastEthernet0/0
     192.168.2.0/30 is subnetted, 1 subnets
B       192.168.2.0 [20/0] via 192.168.1.1, 01:44:36
```

```
CE2#sh ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

     10.0.0.0/32 is subnetted, 2 subnets
B       10.1.10.1 [20/0] via 192.168.2.1, 01:45:51
C       10.2.10.1 is directly connected, Loopback0
     192.168.1.0/30 is subnetted, 1 subnets
B       192.168.1.0 [20/0] via 192.168.2.1, 01:45:51
     192.168.2.0/30 is subnetted, 1 subnets
C       192.168.2.0 is directly connected, FastEthernet0/0
```

Figura 64. Verificación de rutas en SR MPLS
Fuente: Elaboración propia

De igual manera se verifica los saltos y asignación de segmento para SR MPLS y etiqueta para VPNv4, con el comando *traceroute*, se puede apreciar que a lo largo de los saltos la etiqueta asignada al segmento destino no cambia

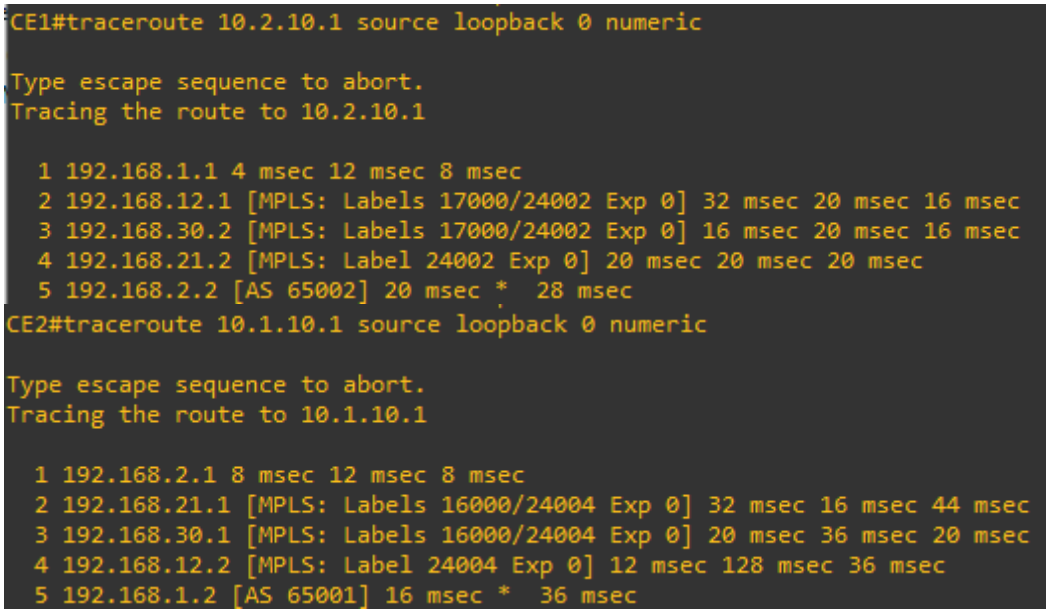


Figura 65. Prueba de traceroute en SR MPLS
Fuente: Elaboración propia

Con la prueba de ping se demuestra la conectividad *end-to-end:*

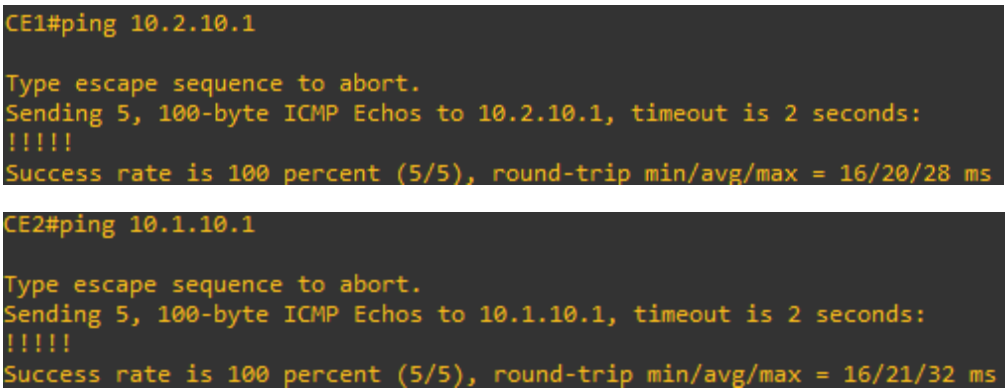


Figura 66. Prueba de ping entre CE en SR MPLS
Fuente: Elaboración propia

4.3 SEGMENT ROUTING PARA IPv6 o SRv6

En SR MPLS una etiqueta se usa como un Segment Identifier SID, donde el router de origen escoge una ruta hacia el destino y la codifica en la cabecera del paquete como un stack de etiquetas. En cambio en SRv6 una dirección IPv6 se usa como SID, donde el router de origen codifica la ruta hacia el destino como una lista ordenada de segmentos o de direcciones IPv6 que se colocan en una nueva cabecera llamada Segment Routing Header SRH. A lo largo de la ruta que un paquete tome en SRv6 existen nodos con diferente funcionalidad:

- **Source node:** El nodo que genera el paquete IPv6 y que coloca su respectivo SRH.
- **Transist node:** Uno nodo que está presente en la ruta de SRv6, pero que no revisa el SRH, es decir la dirección IPv6 de destino no corresponde a este nodo.
- **End point node:** Es el nodo donde termina el segmento SRv6, es decir la dirección IPv6 con su SRH corresponden a este nodo. (CISCO SYSTEMS, 2019)

El SID en SRv6 es una dirección de 128 bits que comprende las siguientes partes:

- **Locator:** Son los 64 bits más significantes y representan la dirección de un nodo SRv6, a su vez esta primera parte puede ser dividida en dos partes:
    - **SID Block** son los 40 bits más significantes de la parte de locator, indica la parte de red de SRv6 que es fija y conocida
    - **Node Id** son los 24 bits menos significantes de la parte de locator, y hace referencia a la parte del nodo de SRv6
- **Function:** Esta parte del SID tiene significado local para el nodo e indica una función SRv6 específica que solo se ejecuta en el nodo local, por ejemplo L3VPNv4.
- **Args:** Es un campo opcional e indica argumentos opcionales, por ejemplo Metadata.

De esta manera es posible realizar Network Program en la cabecera de un paquete, con lo cual la ruta hacia el destino se establece desde el inicio del dominio de SRv6.

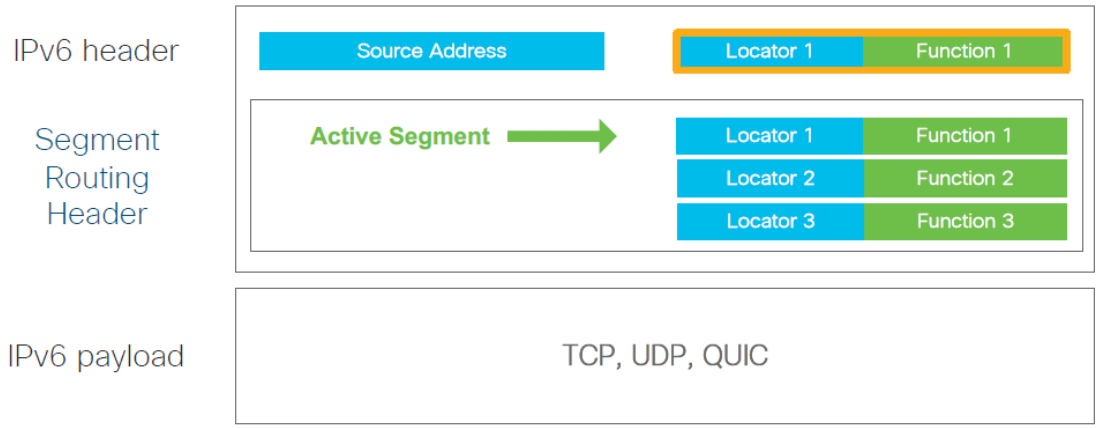


Figura 67. Network Program en SRH
Obtenido de (Camarillo, 2019)

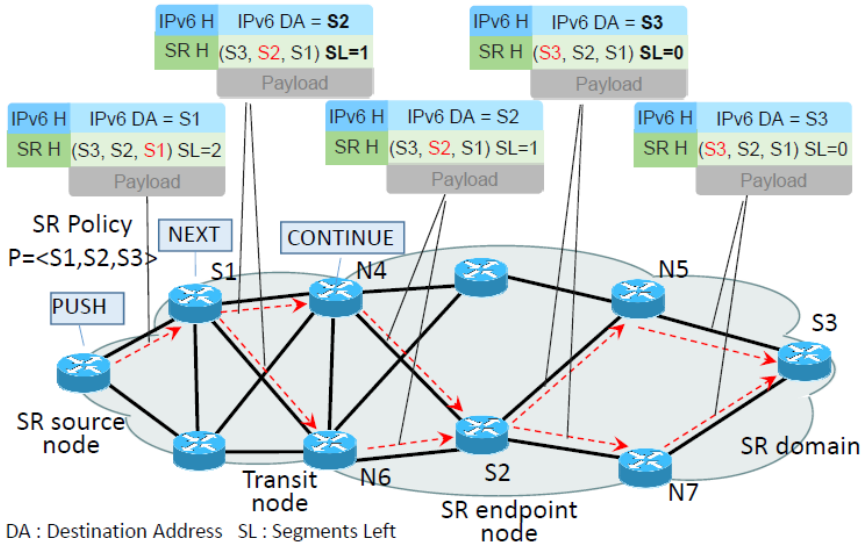


Figura 68. Operación de SRv6 data plane
Obtenido de (Ventre, y otros, 2020)

4.3.1 **Arquitectura SRv6 a simular**

Se sigue manteniendo el escenario de simulación que se ha tratado desde el numeral 4.1.4 y que comprende la comunicación de dos sucursales remotas a través del backbone del SP. Cabe indicar que para poder simular este escenario los equipos deben ser Cisco ASR9000

Series Routers, IOS XR Release 6.6.1 (CISCO SYSTEMS, 2019), en tanto que para este caso de estudio se utiliza la versión actual del emulador que es IOS XR Release 6.0.1, 6.1.3, por lo tanto se expone de forma teórica los comandos a ser utilizados para su futura aplicación real.

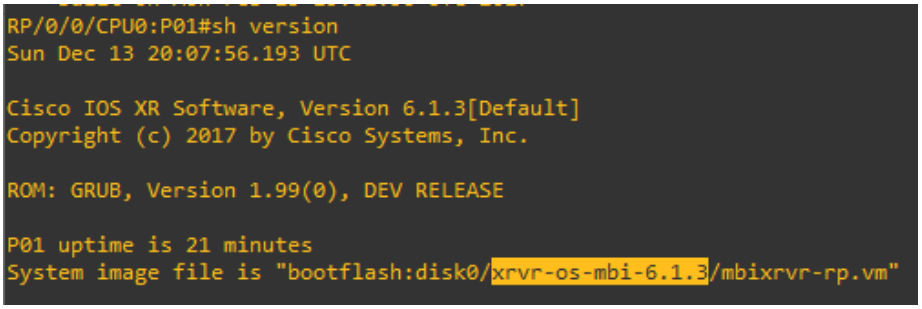


Figura 69. Versión de CISCO IOS XR de equipos emulados
Fuente: Elaboración propia

- **Configuración de SRv6**

Se lo habilita de forma global, para lo cual es necesario configurar un locator con su prefijo, que se anuncia en redes IPv6 y SRv6 por el protocolo IS-IS, teniendo en cuenta que todos los routers dentro del dominio SRv6 deben tener el mismo SID Block, si este valor es menor a 40 bits, se debe rellenar con ceros. Los comandos a utilizar son los siguientes

RP/0/0/CPU0:PE1(config)#segment-routing srv6
RP/0/0/CPU0:PE1(config-srv6)#locators
RP/0/0/CPU0:PE1(config-srv6-locators)#locator *myLoc1*
RP/0/0/CPU0:PE1(config-srv6-locators)#prefix 2001:db8:a1:1::/64

Con el comando **show segment-routing srv6 locator** *Loc1* **detail** se verifica la configuración del locator (CISCO SYSTEMS, 2019), también con el comando **show segment-routing sid**, se puede verificar información de SID.

```
Router# show segment-routing srv6 locator myLoc1 detail
Name                  ID      Prefix                    Status
--------------------- ------- ------------------------- -------
myLoc1*               5       2001:db8:0:a2::/64        Up
  (*): is-default
  Interface:
    Name: srv6-myLoc1
    IFH : 0x00000170
    IPv6 address: 2001:db8:0:a2::/64
  Chkpt Obj ID: 0x2fc8
  Created: Apr 25 06:21:57.077 (00:03:37 ago)
```

Figura 70. Verificación de locator en SRv6
Obtenido de (CISCO SYSTEMS, 2019)

Con los siguientes comandos **show**, se puede comprobar las configuraciones de SRv6 global y locator

Command	Description
show segment-routing srv6 manager	Displays the summary information from SRv6 manager, including platform capabilities.
show segment-routing srv6 locator *locator-name* [**detail**]	Displays the SRv6 locator information on the router.
show segment-routing srv6 locator *locator-name* **sid** [[*sid-ipv6-address* [**detail**]	Displays the information regarding SRv6 local SID(s) allocated from a given locator.
show segment-routing srv6 sid [*sid-ipv6-address* | **all** | **stale**] [**detail**]	Displays SID information across locators. By default, only "active" (i.e. non-stale) SIDs are displayed.
show route ipv6 local-srv6	Displays all SRv6 local-SID prefixes in IPv6 RIB.

Figura 71. Comandos show para SRv6 global y locator
Obtenido de (CISCO SYSTEMS, 2019)

- **Configuración de SRv6 IS-IS**

Este protocolo soporta SR-MPLS, y para trabajar con SRv6 habilita una extensión para soportar los SRv6 SIDs, permitiendo aprender prefijos locales de locator y anunciarlos en el dominio del IGP, de igual manera aprender prefijos remotos de locator e instalarlos en la RIB o FIB y, aprender prefix SID y adjacency SID para anunciarlos en el backbone del IGP. Se debe tomar en cuenta que una address-family de IS-IS puede soportar una sola forma de segment routing, SR-MPLS o SRv6 (CISCO SYSTEMS, 2019). La configuración es la siguiente

```
RP/0/0/CPU0:PE1(config)#router isis 1
RP/0/0/CPU0:PE1(config-isis)#address-family ipv6 unicast
RP/0/0/CPU0:PE1(config-isis-af)# segment-routing srv6
RP/0/0/CPU0:PE1(config-isis-srv6)#locator myLoc1
RP/0/0/CPU0:PE1(config-isis-srv6-loc)#exit
```

- **Configuración de SRv6 para L3VPN**

Para tener una analogía a los dos escenarios anteriores se puede configurar el servicio de IPv4 L3VPN sobre un data plane de SRv6 para permitir la comunicación de varios sitios de una red privada a través una red pública que es el backbone del SP, con el uso de SRv6 Segment IDs (SIDs) en lugar de etiquetas. La comunicación entre CE-PE por VRF es en IPv4, con protocolo BGP por lo que se provee soporte End.DT4/DX4 que es el último punto con des encapsulación, cross-conexión y búsqueda en la tabla IPv4, en tanto que para el backbone del SP en el dominio del IGP se utiliza IPv6, los equipos PE con las respectivas configuraciones de SRv6 para soportar las sesiones MP-BGP de VPNv4, donde se encapsula este tráfico con los SRv6 SID y se lo envía a través del backbone del SP, donde los equipos P tienen también configuraciones de SRv6 en lo referente a SID (CISCO SYSTEMS, 2019); aunque estos últimos también pueden trabajar sólo con IS-IS IPv6 sin necesidad de usar SRv6, de tal manera que el tráfico en los equipos P es tratado como IPv6 normal, permitiendo que esta solución sea efectiva y simple. Cuando un prefijo se asocia con un SID, el equipo PE egress anuncia este SID a través de BGP de VPNv4 hacia el equipo PE ingress que conocerá el prefijo de VPNv4 y el SID que usa; de esta forma el equipo PE ingress va a encapsular el tráfico hacia el equipo PE egress con nuevo IPv6 header cuya dirección de destino es el SID asociado al prefijo VPNv4. (Gonzalez, 2020)

Las configuraciones para permitir el servicio de L3VPNv4 sobre SRv6 son las siguientes:

- Configuración de SRv6 locator dentro de la familia VPNv4

  RP/0/0/CPU0:PE1(config)#router bgp *65000*
  RP/0/0/CPU0:PE1(config-bgp)#bgp router-id *10.1.1.1*
  RP/0/0/CPU0:PE1(config-bgp)# address-family vpnv4 unicast
  RP/0/0/CPU0:PE1(config-bgp-af)#segment-routing srv6
  RP/0/0/CPU0:PE1(config-bgp-af-srv6)#locator *myLoc1*
  RP/0/0/CPU0:PE1(config-bgp-af-srv6)#exit

- Configuración de VRF con alojamiento de etiqueta por VRF

  RP/0/0/CPU0:PE1(config-bgp-af)#vrf *netdat001*
  RP/0/0/CPU0:PE1(config-bgp-vrf)# rd *65000:1*
  RP/0/0/CPU0:PE1(config-bgp-vrf)#address-family ipv4 unicast
  RP/0/0/CPU0:PE1(config-bgp-vrf-af)#segment-routing srv6
  RP/0/0/CPU0:PE1(config-bgp-vrf-af-srv6)#alloc mode per-vrf
  RP/0/0/CPU0:PE1(config-bgp-vrf-af-srv6)#exit
  RP/0/0/CPU0:PE1(config-bgp-vrf-af)#exit
  RP/0/0/CPU0:PE1(config-bgp-vrf)#neighbor 192.168.1.2
  RP/0/0/CPU0:PE1(config-bgp-vrf-nbr)#remote-as 65001
  RP/0/0/CPU0:PE1(config-bgp-vrf-nbr)# address-family ipv4 unicast

A continuación, se presenta la arquitectura para SRv6, y que como se indica al comienzo de este numeral se expone de forma teórica por la imagen IOS XR con que se cuenta.

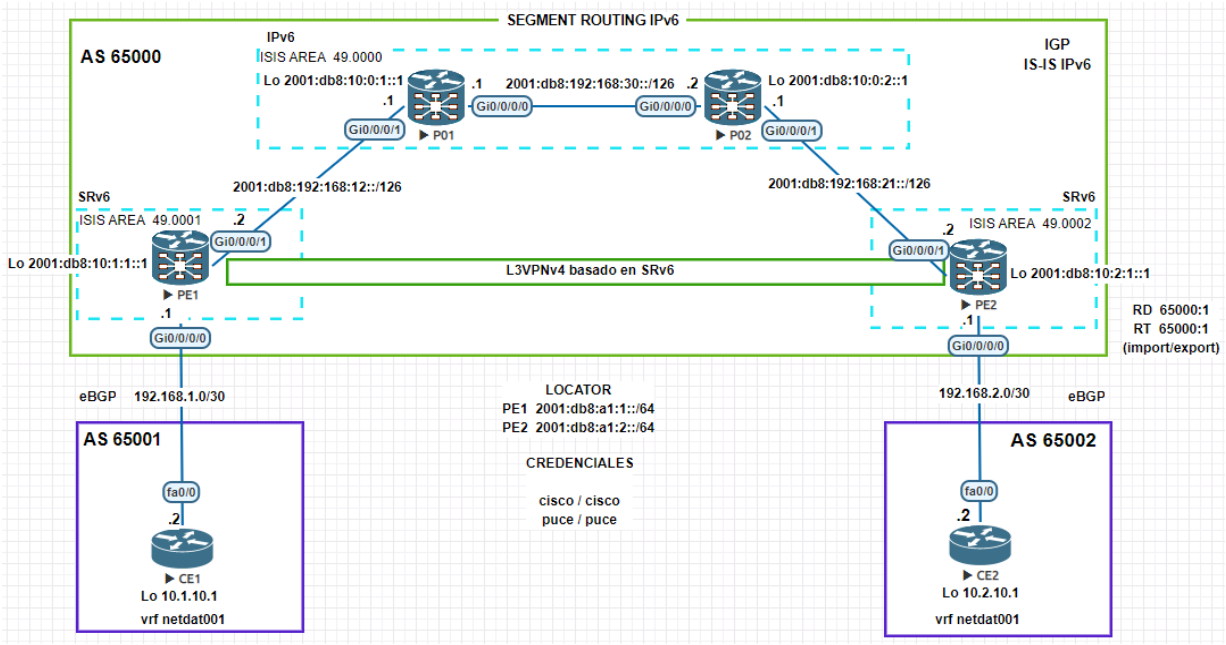


Figura 72. Arquitectura L3VPNv4 basado en SRv6
Fuente: Elaboración propia

Debido a que en el dominio del SP se utiliza IPv6 y en los sitios remotos IPv4, las direcciones son las siguientes:

Tabla 5. Direccionamiento de arquitectura SRv6

EQUIPO	INTERFACE	DIRECCIÓN	AS
CE1	Fa0/0 Lo0	192.168.1.2/30 10.1.10.1/32	65001
CE2	Fa0/0 Lo0	192.168.2.2/30 10.2.10.1/32	65002
PE1	Gi0/0/0/0 Gi0/0/0/1 Lo0 locator	192.168.1.1/30 2001:db8:192:168:12::2/126 2001:db8:10:1:1::1/128 2001:db8:a1:1::/64	65000
PE2	Gi0/0/0/0 Gi0/0/0/1 Lo0 locator	192.168.2.1/30 2001:db8:192:168:21::2/126 2001:db8:10:2:1::1/128 2001:db8:a1:2::/64	
P01	Gi0/0/0/0 Gi0/0/0/1 Lo0	2001:db8:192:168:30::1/126 2001:db8:192:168:12::1/126 2001:db8:10:0:1::1/128	
P02	Gi0/0/0/0 Gi0/0/0/1 Lo0	2001:db8:192:168:30::2/126 2001:192:168:21::1/126 2001:db8:10:0:2::1/128	

Fuente: Elaboración propia

Hay que recordar que este escenario requiere de contar con la imagen IOS XR Release 6.6.1, lo que fue un reto para el proceso de emulación, por ello, se presentan las pruebas que se pueden ejecutar para comprobar el funcionamiento de SRv6.

- **Verificación de información generada de SRv6**

Una vez levantado el entorno de SRv6, se puede comprobar la asignación de locator, cabe indicar que End.DT4 es el punto final donde se des encapsula y se busca la tabla IPv4 por cada *vrf*. Se usa los comandos *show segment-routing srv6 sid* y *show segment-routing srv6 sid* ***sid-prefix*** *detail con lo que* se comprueba la asignación y estado de un sid:

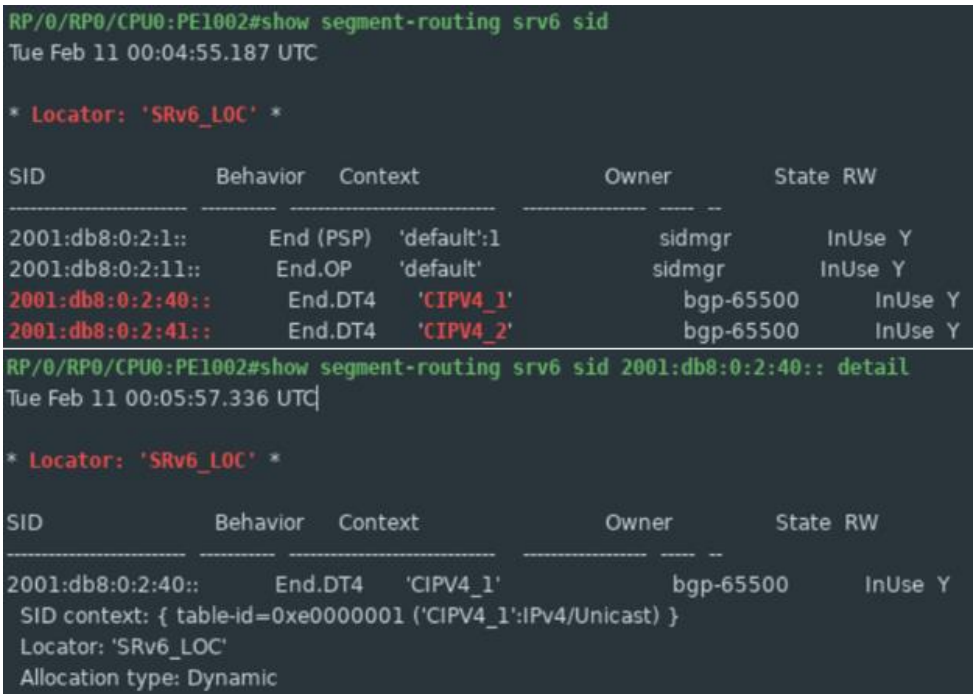


Figura 73. Verificación de SRv6 SID
Obtenido de (Gonzalez, 2020)

- **Verificación de SRv6 SID para VPNv4**

Es posible consultar los prefijos que se aprenden en el equipo PE para una *vrf*, y por tanto la dirección SRv6-VPN-SID que se asigna a dicha *vrf*. Con los comandos *show bgp vpnv4 vrf* ***vrf-name*** y *show bgp vpnv4 unicast vrf* ***vrf-name prefix*** se obtiene dicha información:

Figura 74. Prefijos aprendidos para una vrf en SRv6
Obtenido de (Gonzalez, 2020)

- **Verificación de data plane para una VRF**

Para verificar que el tráfico de una *vrf* de IPv4 es transportada sobre el data plane de SRv6 se lo realiza con el comando *show cef* ***vrf-name prefix***:

Figura 75. Data plane de SRv6 para un prefijo VPNv4
Obtenido de (Gonzalez, 2020)

- **Verificación de conectividad en SRv6**

Para comprobar que el escenario de L3VPNv4 basado en SRv6 funciona se puede realizar un ping entre equipos CE y para comprender este funcionamiento a continuación se muestra la forma que en un paquete IPv4 viaja desde un CE1 hacia otro CE2 y viceversa a través del backbone del SP, indicando las direcciones que se asignan en los equipos PE donde se realiza el soporte de una VPNv4 sobre SRv6.

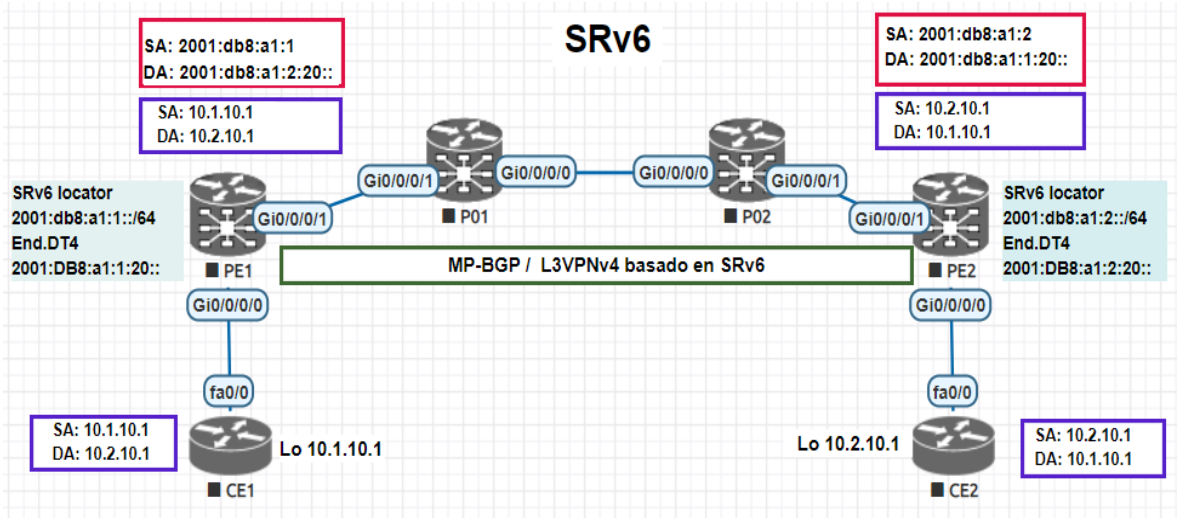


Figura 76. Conectividad end-to-end en SRv6
Fuente: Elaboración propia

4.4 ANÁLISIS DE RESULTADOS

En los escenarios descritos en este capítulo se observa el modo de operación de cada arquitectura con sus respectivas características, y a continuación se establece una comparativa de estas arquitecturas a fin de determinar las ventajas de SRv6 sobre las otras.

Tabla 6. Análisis comparativo MPLS vs SR MPLS vs SRv6

MPLS	SR MPLS	SRv6
ETIQUETADO DE TRÁFICO EN TRÁNSITO		
Mediante Labels Label Stack Topmost Label	Mediante SIDs SR Policy Active Segment	Mediante IPv6 address Segment List en SRH IPv6 address en Destination Address.
OPERACIONES		
Push	Push	Símil SR MPLS, pero añadiendo IPv6 a Segment List en SRH.
Swap	Continue	Enviar de acuerdo a la IPv6 Destination Address.
Pop	Next	Decrementa Segment Left y se copia el active segment en IPv6 Destination Address.

SOPORTE DE TECNOLOGÍA		
Actualmente, todos los equipos de red modernos lo soportan. Tecnología en operación con gran soporte técnico, como por ejemplo MPLS L3VPN.	Requiere actualización de software, ya que usa el data plane de MPLS. Actualización de conocimiento en SR para operar sobre el data plane de MPLS.	Necesita cambio de hardware y actualización de software para soportar data plane de SRv6. Requiere estudio de nuevos conceptos sobre SRv6 y su despliegue en el backbone del SP es puro en IPv6.
CARACTERÍSTICAS DE LA TECNOLOGÍA		
Número de estados aumenta con el número de rutas	Tabla LFIB se mantiene constante, indiferente de las rutas en una topología Full-Mesh; por lo cual reduce considerablemente el número de estados.	
Utiliza protocolo LDP para distribución de etiquetas y establecimiento de rutas, así como RSVP para optimización de rutas de TE.	No usa LDP, ya que un segmento es una instrucción, usa el concepto de prefix SID, aprovechando el data plane de MPLS.	Se basa en SRv6 SID, donde la ruta se codifica en una lista ordenada de segmentos o IPv6 address que se colocan en el SRH. No utiliza RSVP ni tunneling, su algoritmo es nativo de acuerdo a source routing.
Soporta IPv4 / IPv6	Soporta IPv4 / IPv6	Soporta IPv6

Utiliza la característica de LDP NSR (*Non Stop Routing*), que brinda un mecanismo de rápida recuperación en control plane frente a fallos, para asegurar alta disponibilidad.	50msec en protección por prefijo contra fallos de enlaces, nodos con el uso de TI-LFA (*Topology Independence-Loop Free Alternate*). Se soporta también en escenarios SR/LDP. Se calcula por los routers durante el proceso de IGP. Cubre el 100% de cualquier topología.
Soporta QoS en clasificación de paquetes, prevención de congestiones, con el uso de los bits del campo experimental.	Satisface requerimientos de QoS al asegurar ancho de banda aplicado en el atributo de AB de los segmentos de nodo o adyacencia en la ruta de Segment Routing Traffic Engineering.

Fuente: Elaboración propia

Como se puede apreciar, prevalece lo simple de esta nueva tecnología, y esto gracias al hecho que SRv6 basa su operación en el source routing que permite programar una ruta en la cabecera SRH con segmentos o direcciones IPv6, con lo cual se tiene un data plane de IPv6 rápido y que puede llevar tráfico de IPv6, IPv4 o tramas de capa 2. Su despliegue va a permitir en un futuro cercano apalancar otras tecnologías que demandan gran rapidez de conmutación y procesamiento de información del Core de un service provider, y que

se puede extender a otras aplicaciones Bussiness, Redes de Acceso, Data Center, Content Delivery Networks.

En la siguiente figura se aprecia el despliegue que está teniendo SRv6 a nivel mundial, con dos casos implementados y otros en planificación de despliegue; queda así evidenciado que esta tecnología emergente es quien va a tomar la posta a su predecesor MPLS en un futuro cercano.

Figura 77. Despliegue mundial de SRv6
Obtenido de (Camarillo, 2019)

5 CONCLUSIONES Y RECOMENDACIONES

5.1 CONCLUSIONES

- Se concluye que durante el desarrollo del presente caso de estudio, se ha podido evidenciar a través de la emulación y evaluación de resultados que, la implementación y adopción de Segment Routing, tanto en sus presentaciones de SR-MPLS y SRv6, es factible en ambientes de service providers, con el fin de soportar nuevas tecnologías que requieren altas velocidades de conmutación y gran procesamiento de información.

- Se concluye que MPLS es la tecnología utilizada para la conmutación de paquetes en el core de los service providers en la actualidad, tecnología que ha brindado por más de una década el soporte necesario en el manejo y evolución de tráfico que cada día demanda más contenido multimedia y en tiempo real, lo cual hace necesario pensar en la transición hacia una nueva tecnología que brinde las capacidades para operar con grandes flujos de información, siendo SR MPLS la mejora que aprovecha el mismo data plane para optimizar el funcionamiento del actual MPLS, lo que considero como un tecnología base para el proceso de transición a SRv6 sin mayores cambios en hardware.

- Luego de un proceso de investigación y análisis, se pudo concluir que SRv6 basa su funcionamiento en el concepto de enrutamiento en el origen, por tanto, simplifica el enrutamiento y procesamiento dentro del backbone del service provider al brindar una solución simple, escalable y efectiva, a través del network programming desde el

origen que inscribe la ruta hacia el destino por medio de direcciones IPv6 en la cabecera SRH.

- Con la emulación de la arquitectura de red en el escenario de una L3VPN que conecta dos sucursales remotas a través del backbone de un service provider, se ha podido evidenciar y concluir que, la operación de MPLS se realiza a través del protocolo LDP con la asignación de etiquetas al pasar por cada nodo; mientras que SR MPLS del escenario 2, aprovecha el data plane de MPLS, eliminando LDP, introduciendo los prefix SID e inscribiendo una lista ordenada de instrucciones o SR Policy colocada en la cabecera del paquete, y SRv6 del escenario 3, establece un backbone puro sobre un data plane IPv6 que se basa en el enrutamiento desde el origen mediante la inscripción de direcciones IPv6 que se forman en base a los SRv6 SID que se colocan en la cabecera SRH, con lo cual el enrutamiento dentro del service provider se vuelve simple, efectivo y rápido.

- SRv6 potencia y mejora características de latencia, jitter y QoS al asegurar ancho de banda que se aplica a los atributos de los segmentos de nodo o adyacencia en una ruta de Segment Routing Traffic Engineering SR-TE y que no utiliza el protocolo RSVP, así también mejora el Fast Re-Route (FRR) para recuperación ante fallas, soporta VPN que tienden a ser IPv6 hasta los clientes, y la automatización de red con Network Function Virtualization y Software Defined Networking.

- SR MPLS, requiere una actualización en software, siendo una mejora para MPLS, ya que optimiza la infraestructura actual, en tanto que SRv6 requiere un cambio en hardware lo cual representa inversión para los operadores así como capacitación en el talento humano, pero, el beneficio de estos cambios va a generar gran interés en los operadores al permitir soportar nuevas aplicaciones de red que demandan gran cantidad de procesamiento de información como redes de acceso FTTx, core, data centers, 5G, IoT entre otros, y que tienden a migrar hacia el entorno IPv6 en dispositivos finales.

- Si a los equipos P (Provider) dentro del dominio de SRv6 se los trata como *transit node*, y a los equipos PE (Provider Edge) como *source node* o *end point node*, se obtiene una solución más simple de SRv6 ya que al interior del backbone de un service provider el tráfico se va a tratar como IPv6 normal, reduciendo el tamaño de la cabecera SRH y aportando mayor velocidad de conmutación al interior del backbone del service provider.

- SRv6 al ser una tecnología emergente y con pronta aplicación a futuro tiene su grupo de desarrollo y constante investigación que es SPRING WG Source Packet Routing in Networking Working Group, siendo la RFC 8402, la publicación más representativa sobre la arquitectura Segment Routing.

5.2 RECOMENDACIONES

- Para la emulación de los escenarios del presente trabajo se recomienda al menos contar con 16GB de RAM dado que las imágenes de los equipos XR consumen 3GB cada uno, en tanto que los equipos IOS consumen 512MB, así como el uso de 4vCPU. De igual manera para no sobrecargar a la máquina local en la que se realice la emulación, es muy recomendable montar EVE-NG sobre un cloud de libre elección, para el caso de la presente tesis se usó Google Cloud, con lo cual se aprovecha las ventajas de una IaaS *Infraestructure as a Service* con alta disponibilidad de recursos de hardware y uso bajo demanda.

- Para poder emular el escenario de SRv6 como prueba de concepto a futuro, se recomienda utilizar la versión de IOS XR Release 6.6.1 o superior, así también puede desarrollarse sobre otros softwares como Linux kernel o Cumulus Linux.

- En los equipos de arquitectura CISCO XR se recomienda aplicar un *route policy* en la configuración de BGP para que puedan pasar rutas en direcciones de entrada y salida, caso contrario no se va a permitir el intercambio de prefijos a través de BGP hacia los CE como MP-BPG para las L3VPN.

- Para futuros casos de estudio basados en este trabajo se puede realizar la prueba de funcionamiento de Segment Routing con OSPF como protocolo de IGP en lugar de IS-IS, dentro del backbone del service provider. También se puede realizar pruebas de concepto de SRv6 con SD-WAN.

6 BIBLIOGRAFÍA

Acosta, A., & Fonseca, J. (28-31 de Mayo de 2019). *Pregunte al Experto- Introducción a Segment Routing IPv6(SRv6).* Obtenido de https://community.cisco.com/t5/discusiones-routing-y-switching/pregunte-al-experto-introducci%C3%B3n-a-segmnet-routing-ipv6-srv6/td-p/3863614

Behfor (Dirección). (2018). *[HOWTO] Test My Network Speed?! [iPerf & JPerf]* [Película].

Cabrera, C. (06 de Mayo de 2018). *Qué es EVE-NG? el «nuevo» emulador para redes.* Obtenido de https://cesarcabrera.info/que-es-eve-ng-un-nuevo-emulador-para-redes/

Cai, D., Wielosz, A., & Wei, S. (19 de Junio de 2014). Evolve carrier ethernet architecture with SDN and segment routing. *IEEE*.

Camarillo, P. (16 de Mayo de 2019). Srv6 Network Programming. Madrid, Madrid, España.

Capillas, M. (23 de Diciembre de 2016). *La evolución de las redes WAN pasa por la tecnología SDN.* Obtenido de https://empresas.blogthinkbig.com/la-evolucion-de-las-redes-wan-pasa-por-la-tecnologia-sdn/

CISCO. (2014). *Implementing Cisco Service Provider Next-Generation Core Network Services* (Vol. 1). San Jose, California, Estados Unidos.

CISCO. (9 de Agosto de 2018). Introduction to Cisco Segment Routing Traffic Engineering. Obtenido de https://www.youtube.com/watch?v=XZW5wHwhXoE

Cisco Comunity. (30 de Mayo de 2019). *Spanish Webcast- Introducción a Segment Routing IPv6 (SRv6).* Obtenido de https://www.youtube.com/watch?v=OS0fct5QKww

Cisco Networking Academy. (2014). Connecting Networks. En C. N. Academy, *Connecting Networks* (págs. 104-124).

CISCO SYSTEMS. (30 de 04 de 2019). Segment Routing Configuration Guide for Cisco ASR 9000 Series Routers, IOS XR Release 6.6.x. San Jose, California, USA.

Deering, S., & Hinden, R. (Diciembre de 1998). *Internet Protocol, Version 6 (IPv6) Specification*. Obtenido de https://tools.ietf.org/html/rfc2460

Don, J. (06 de Septiembre de 2018). *Fundamentos de la tecnología WDM: Diferencia entre la tecnología CWDM y DWDM.* Obtenido de https://medium.com/@xxxamin1314/fundamentos-de-la-tecnolog%C3%ADa-wdm-diferencia-entre-la-tecnolog%C3%ADa-cwdm-y-dwdm-9ed16b22a0a9

Duchene, F., Jadin, M., & Bonaventure, O. (20 de Agosto de 2018). Exploring various use cases for IPv6 Segment Routing. (ACM, Ed.) *SIGCOMM '18*, 129-131.

EVE-NG - Emulated Virtual Environment. (20 de Marzo de 2017). *EVE install Telnet VNC Wireshark Local management*. Obtenido de https://www.youtube.com/watch?v=Ea4U93991dw

EVE-NG. (18 de Octubre de 2019). *EVE-The Emulated Virtual Environment for Network, Security and DevOps professionals*. Obtenido de https://www.eve-ng.net/

Filsfils, C., Kumar Nainar, N., Pignataro, C., Cardona, J. C., & Francois, F. (06 de Diciembre de 2015). The Segment Routing Architecture. *2015 IEEE Global Communications Conference (GLOBECOM).* San Diego, CA, USA: IEEE.

Gafni, B. (23 de Octubre de 2018). *On Segment(ed) Routing.* Obtenido de https://blog.mellanox.com/2018/10/segment-routing-using-mpls-ipv6-srv6/

Gonzalez, A. (11 de Febrero de 2020). *SRv6 a new Hope. Case 1: SRv6-Based IPv4 L3VPN*. Obtenido de https://www.linkedin.com/pulse/srv6-new-hope-case-1-srv6-based-ipv4-l3vpn-asier-gonzalez-diaz/

Google. (10 de Enero de 2021). *IPv6 Adoption*. Obtenido de https://www.google.com/intl/en/ipv6/statistics.html

Google. (10 de Enero de 2021). *Per-Country IPv6 adoption*. Obtenido de https://www.google.com/intl/en/ipv6/statistics.html#tab=per-country-ipv6-adoption

Google Code Archive. (s.f.). *xjperf*. Obtenido de https://code.google.com/archive/p/xjperf/downloads

Internet Engineering Task Force (IETF). (Julio de 2018). *Segment Routing Architecture.* Obtenido de https://tools.ietf.org/html/rfc8402

iPerf. (2019). *iPerf - The ultimate speed test tool for TCP, UDP and SCTP*. Obtenido de https://iperf.fr/

iPerf.fr. (Enero de 2020). *iPerf - The ultimate speed test tool for TCP, UDP and SCTP*. Obtenido de https://iperf.fr/

Jaksic, D. (19 de Marzo de 2018). *Segment Routing in Service Provider networks* . Obtenido de https://www.cisco.com/c/dam/m/hr_hr/training-events/2018/cisco-connect/pdf/Segment_Routing_in_Service_Provider_Network_-_Dejan_Jaksic.pdf

Juniper Networks. (2020). *NorthStar Controller*. Obtenido de https://www.juniper.net/us/en/products-services/sdn/northstar-network-controller/

juuncaal. (2019). *evolucion de internet*. Obtenido de https://www.timetoast.com/timelines/evolucion-de-internet-285e2ce7-cf55-495d-ba56-d8b0274d0b77

Li, Z., & Wu, N. (9 de Marzo de 2015). *Bandwidth-Guaranteed Segment Routing draft-li-spring-bw-guaranteed-sr-00.* Obtenido de https://tools.ietf.org/html/draft-li-spring-bw-guaranteed-sr-00

Marrone, L., & Salazar, G. (Febrero de 2019). *NUEVA GENERACION DE REDES DE TELECOMUNICACIONES.* Obtenido de https://postgrado.info.unlp.edu.ar/wp-content/uploads/2019/02/2019-Nueva-generacion-de-Redes-de-Telecomunicaciones.pdf

Naranjo, E. F., & Salazar Ch., G. D. (2017). Underlay and overlay networks: The approach to solve addressing and segmentation problems in the new networking era: VXLAN encapsulation with Cisco and open source networks. *IEEE Second Ecuador Technical Chapters Meeting (ETCM)*, 1-6. doi:10.1109/ETCM.2017.8247505

Parada Visual. (29 de 11 de 2019). *¿Qué es SD-WAN definida por software?* Obtenido de https://www.paradavisual.com/que-es-sd-wan-definida-por-software/

Practonet. (s.f.). *Multiplexing – Definition – Types of Multiplexing: FDM, WDM, TDM.* Obtenido de http://practonet.com/ftth/What_is_multiplexing_types_of_multiplexing.php

Salazar Ch., G. D., & Chafla, G. X. (Junio de 2015). Empleo de path-control tools en una red empresarial moderna mediante políticas de enrutamiento. *3C Tecnologia, 4*(1), 1-18. Obtenido de http://ojs.3ciencias.com/index.php/3c-tecnologia/article/view/233

Salazar Ch., G. D., Naranjo, E., & Marrone, L. (8-10 de Noviembre de 2018). SDN-Ready WAN networks: Segment Routing in MPLS-Based Environments. (IEEE, Ed.) *2018 9th IEEE Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)*, 173-178. doi:10.1109/UEMCON.2018.8796613

Salazar Ch., G. D., Venegas, C., & Marrone, L. (2019). MQTT-Based Prototype Rover with Vision-As-A-Service (VAAS)in an IoT Dual-Stack Scenario. *2019 Sixth International Conference on eDemocracy & eGovernment (ICEDEG)*, 344-349. doi:10.1109/ICEDEG.2019.8734341

Salazar Ch., G. D., Venegas, C., Baca, M., Rodríguez, I., & Marrone, L. (2018). Open Middleware proposal for IoT focused on Industry 4.0. (IEEE, Ed.) *2018 IEEE 2nd Colombian Conference on Robotics and Automation (CCRA)*, 1-6. doi:10.1109/CCRA.2018.8588117

Salazar, G. (2 de Febrero de 2016). *Direccionamiento IPv6 - Bases y Fundamentos*. (Cisco, Editor) Obtenido de https://supportforums.cisco.com/blog/12914981/direccionamiento-ipv6-bases-y-fundamentos.

Salazar, G. (25 de Septiembre de 2016). *Fundamentos de QoS -Calidad de Servicio en Capa 2 y Capa 3*. Obtenido de https://community.cisco.com/t5/blogs-routing-y-switching/fundamentos-de-qos-calidad-de-servicio-en-capa-2-y-capa-3/ba-p/3103715

Salazar, G. (25 de Octubre de 2019). Comunicaciones Unificadas y VoIP - PUCE. *Unidad 1, 2. Material no publicado*. Quito, Pichincha, Ecuador.

Santos, V. (03 de Abril de 2019). *Simplifique su red IP con Segment Routing, una parte de Adaptive IP de Ciena*. Obtenido de https://www.ciena.com.mx/insights/articles/Simplify-your-IP-network-with-centralized-Segment-Routing-part-of-Cienas-Adaptive-IP_es_LA.html

TELCEL BELLSOUTH. (s.f.). *¿Qué es Frame Relay?* Obtenido de http://www.geocities.ws/redes_computadoras2_unlm/0377/frame.pdf

The Cisco Learning Network. (2018). *Introduction to Segment Routing.* (D. P, Ed.) Obtenido de https://learningnetwork.cisco.com/s/blogs/a0D3i000002SKP6EAO/introduction-to-segment-routing

Universidad Publica de Navarra. (s.f.). *Introducción a tecnologías WAN.* Obtenido de https://www.tlm.unavarra.es/~daniel/docencia/ftpr/ftpr14_15/slides/Tema2-01-IntroWAN.pdf

Ventre, P. L., Salsano, S., Poverini, M., Cianfrani, A., Abdessalam, A., Filsfils, C., . . . Clad, F. (Junio de 2020). Segment Routing: a Comprehensive Survey of Research Activities, Standardization Efforts and Implementation Results.

Victor, M. (30 de Julio de 2013). *Conmutación de Circuitos.* Obtenido de https://es.slideshare.net/VictorMiles/exposicion-24766631

Vinueza, H. (Julio de 2019). DISEÑO DE UNA RED IP MPLS UTILIZANDO LA ARQUITECTURA SEAMLESS PARA UN PROVEEDOR DE SERVICIOS DE TELECOMUNICACIONES CON COBERTURA EN LA REGION 3 DE ECUADOR. Riobamba, Chimborazo, Ecuador.

7 ANEXOS

7.1 ANEXO 1 – INSTALACIÓN Y PUESTA EN OPERACIÓN DE EVE-NG

A continuación, se describen los pasos básicos para instalación del emulador EVE-NG y carga de imágenes de equipos a simular.

EVE-NG está disponible como archivos ISO y OVA (*Open Virtual Appliance*), siendo éste último formato el que se utiliza en el presente trabajo ya que se monta sobre una máquina virtual como VMware, para entornos empresariales y de mayor demanda EVE-NG se puede instalar sobre un servidor físico dedicado para esta actividad a fin de obtener el mayor desempeño.

Los requerimientos para montar EVE-NG en un entorno virtualizado son:

- Intel CPU VT-x/EPT
- Cualquiera de los siguiente hipervisores:
    - Ubuntu Xenial Xerus 16.04.X LTS 64bit.
    - VMware ESXi 6.0 o posterior
    - VMware Workstation 12.5 o posterior
    - VMware Fusion 8 o posterior
    - VMware Player 12.5 o posterior
    - Google Cloud platform VM

Los requerimientos de CPU y RAM dependen de la cantidad de nodos que el usuario desea emular. Para correr imágenes de IOU/IOL y Dynamips basta con tener 4 vCPU y 6GB de RAM, pero será insuficiente para trabajar con imágenes de routers CSR1000V, donde se requiere desde 3GB de RAM por cada nodo.

7.1.1 INSTALACIÓN DE MÁQUINA VIRTUAL

Para el montaje de EVE-NG, el hipervisor elegido es VMware Workstation en la versión 15.5.0 que puede ser descargado de Internet para uso no comercial, con fines de estudio, como se muestra en las siguientes imágenes.

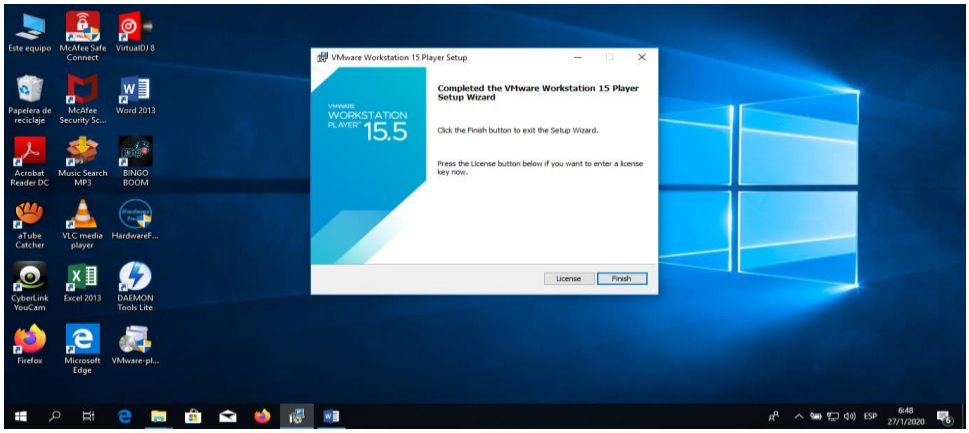


Figura 78. Instalación de VMWare
Fuente: Elaboración propia

7.1.2 INSTALACIÓN DE EVE-NG

Desde la página oficial de EVE-NG https://www.eve-ng.net/index.php/download/#DL-COMM, se descarga la máquina virtual en archivo .OVA

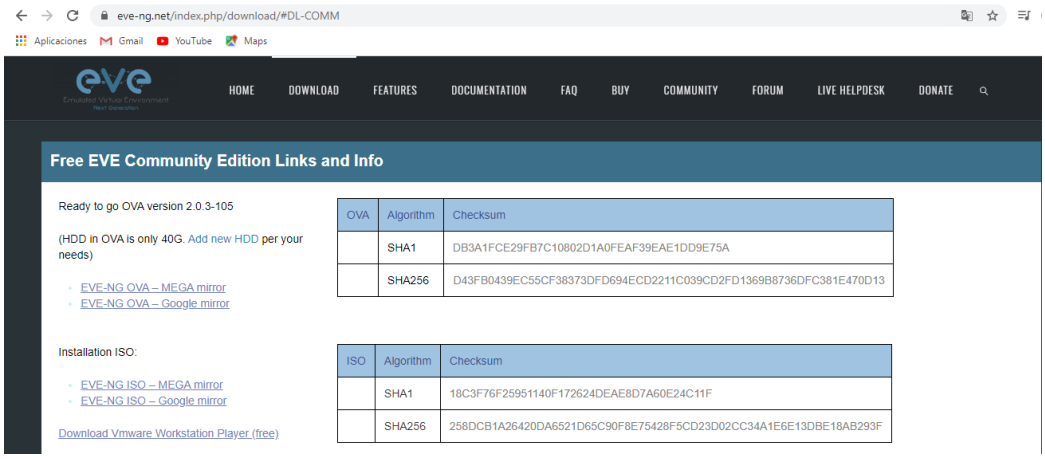


Figura 79. Descarga de EVE-NG
Obtenido de (EVE-NG, 2019)

Adicional, se puede mostrar la opción de descargar el paquete de aplicaciones para cliente utilizados al construir laboratorios

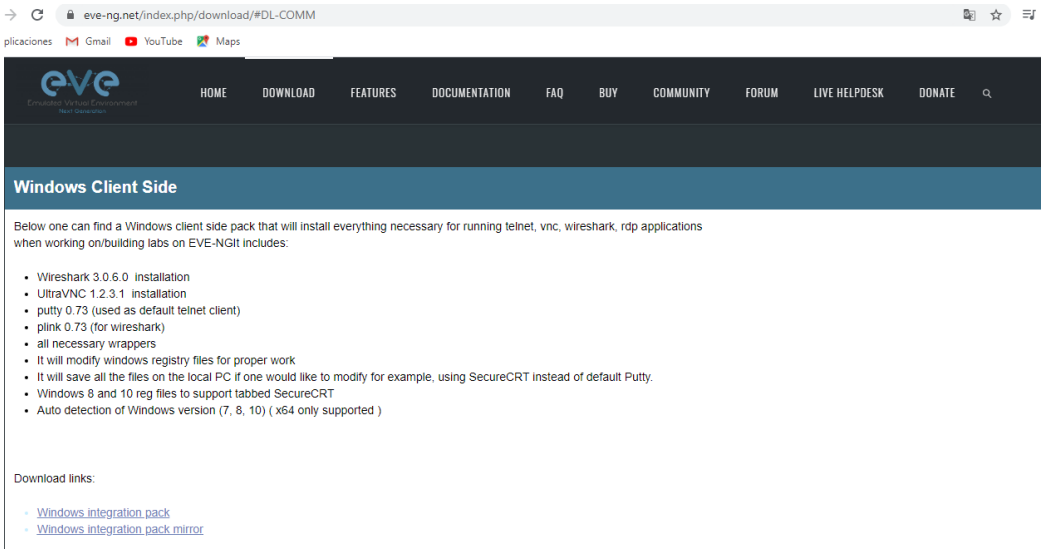


Figura 80. Paquete de aplicaciones para cliente en EVE-NG
Obtenido de (EVE-NG, 2019)

Cuando se tiene el archivo .OVA descargado, se procede a instalar mediante VMware, en la opción de Open a Virtual Machine, ubicando el archivo en el directorio respectivo.

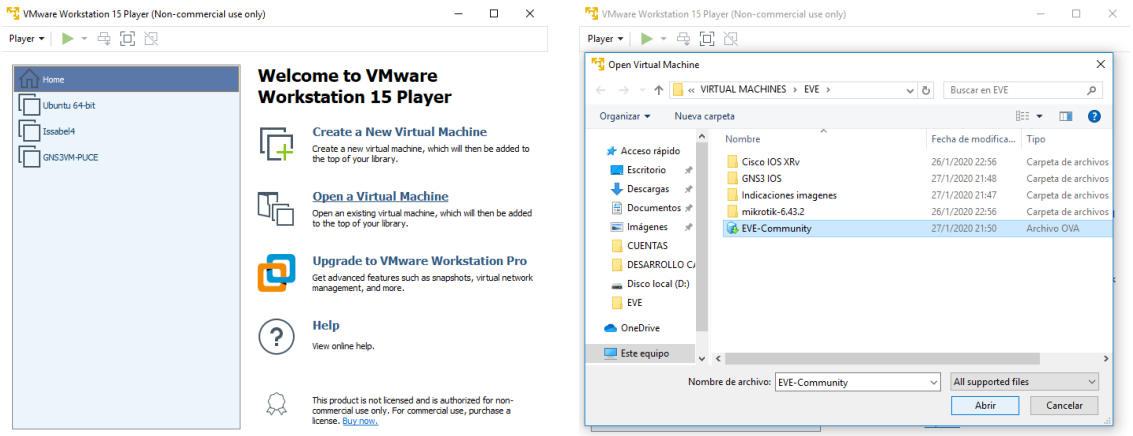


Figura 81. Instalación de EVE-NG parte 1
Fuente: Elaboración propia

Se proporciona un nombre para esta Máquina Virtual y se elige su directorio.

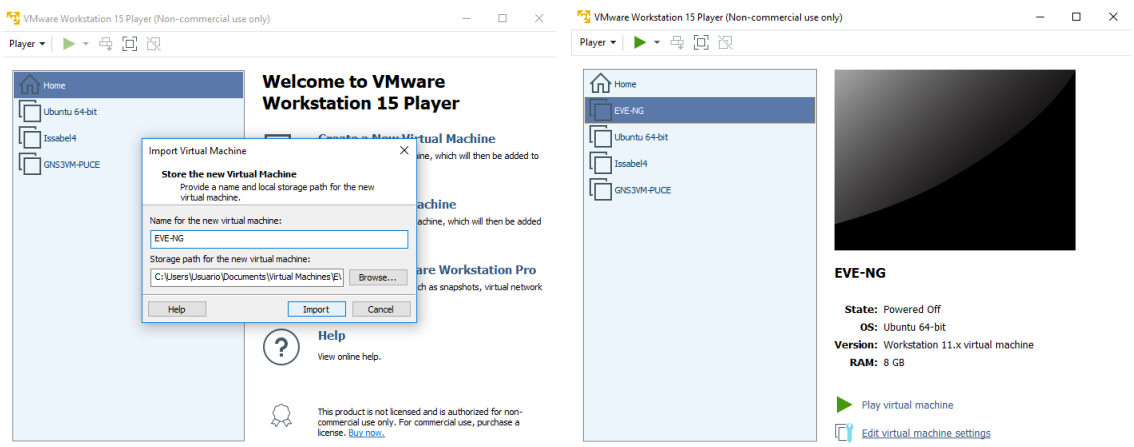


Figura 82. Instalación de EVE-NG parte 2
Fuente: Elaboración propia

A continuación, se asigna la cantidad de memoria necesaria que se requiere por el usuario, se toma en consideración que se va a trabajar con equipos CISCO de arquitectura XR con un consumo promedio de 3GB de RAM por equipo. Se configura también la tarjeta de red para que quede en modo Bridge, esto con el fin de ingresar al entorno de EVE-NG mediante web

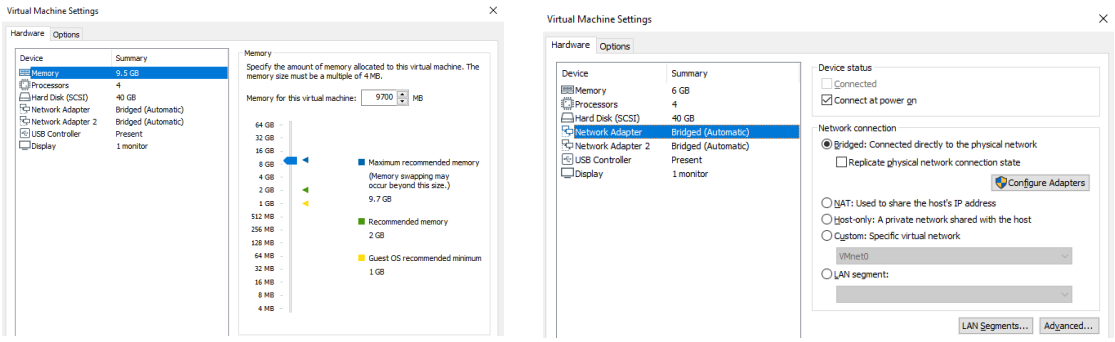


Figura 83. Instalación de EVE-NG parte 3
Fuente: Elaboración propia

Se prende la máquina virtual y se ingresa mediante la dirección IP proporcionada a través de un explorador web

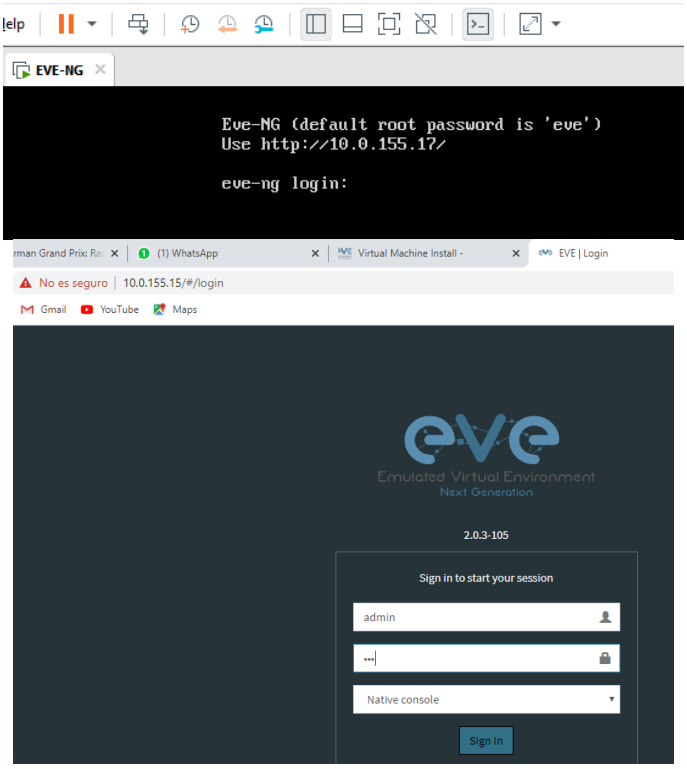


Figura 84. Acceso web a EVE-NG
Fuente: Elaboración propia

Se coloca las credenciales dadas de fábrica y se accede al entorno de trabajo

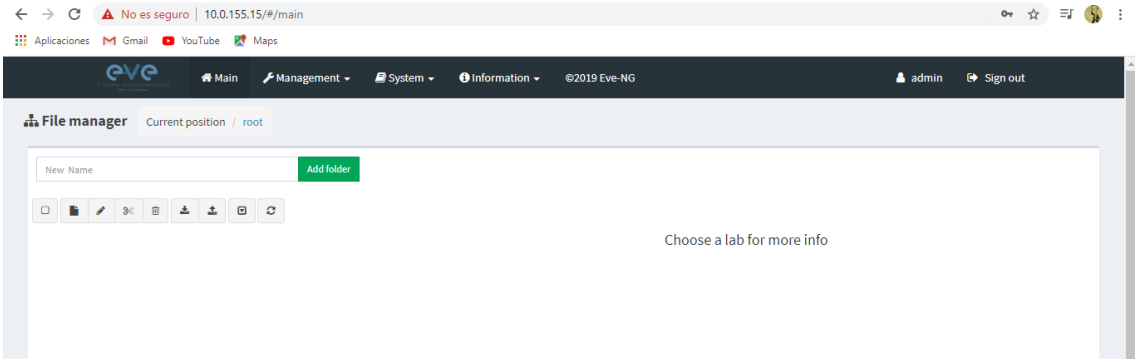


Figura 85. Entorno de trabajo de EVE-NG
Fuente: Elaboración propia

El procedimiento descrito anteriormente se puede complementar con el siguiente enlace provisto en la página web de EVE-NG

https://www.eve-ng.net/index.php/documentation/installation/virtual-machine-install/

7.1.3 CARGAR IMÁGENES EN DYNAMIPS

Dynamips es un motor que permite emular imágenes de routers Cisco de las familias 1700, 2600, 3600, 3700 y 7200, en base a los IOS respectivos que para este emulador tienen asociado un valor Idle PC específico y que se usa para mapear las instrucciones idle del IOS simulado con las instrucciones idle del emulador a través de la plataforma UnetLab; el objetivo es asignar un correcto valor de Idle PC a fin de que el uso de CPU sea menor a 10% para cada nodo simulado. Cabe indicar que EVE-NG permite simular equipos de diferentes fabricantes como Cisco, Juniper, Mikrotik, Huawei, Windows, entre otros. A continuación se describe en breve los pasos para añadir imágenes de Cisco IOS en EVE-NG.

- **IMÁGENES SOPORTADAS EN DYNAMIPS** En la página oficial de EVE-NG se muestran las imágenes de Cisco IOS que pueden simularse y son las del siguiente cuadro, donde se indica el archivo con el cuál se debe descargar, así como el tamaño de memoria virtual RAM y el valor de Idle PC que más adelante se indica cómo asignar la mejor opción:

Tabla 7. Imágenes soportadas en Dynamips

Imagen EVE	Archivo de descarga	Versión	vRAM	Idle PC
c1710-bk9no3r2sy-mz.124-23.image	c1710-bk9no3r2sy-mz.124-23.bin	C1710-BK9NO3R2SY-M 12.4(23)	96	0x80369ac4
c3725-adventerprisek9-mz.124-15.T14.image	c3725-adventerprisek9-mz.124-15.T14.bin	C3725-ADVENTERPRISEK9-M 12.4(15)T14	256	0x60c08728
c7200-adventerprisek9-mz.152-4.S2.image	c7200-adventerprisek9-mz.152-4.S2.bin	C7200-ADVENTERPRISEK9-M 15.2(4)S2	512	0x60630d5c
c7200-adventerprisek9-mz.152-4.S6.image	c7200-adventerprisek9-mz.152-4.S6.bin	C7200-ADVENTERPRISEK9-M 15.2(4)S6		

Fuente: (EVE-NG, 2019)

- **DESCARGA DE IMÁGENES** Se puede realizar la descarga del sitio que contiene las imágenes de Cisco IOS desde https://mega.nz/#F!fpxnXIKB!twpa-jzH4ReWZFq5uZENZg
- **IMPORTACION DE IMÁGENES A EVE-NG** Luego de descargar las imágenes soportadas, las mismas se pueden subir a EVE-NG mediante un programa para transferir archivos como WinSCP. También es necesario ingresar mediante SSH a EVE-NG para realizar las siguientes acciones:

  Crear un directorio temporal en una ubicación fácil de localizar, mediante el comando **mkdir**, para este ejemplo el directorio se llama imágenes_descargadas.

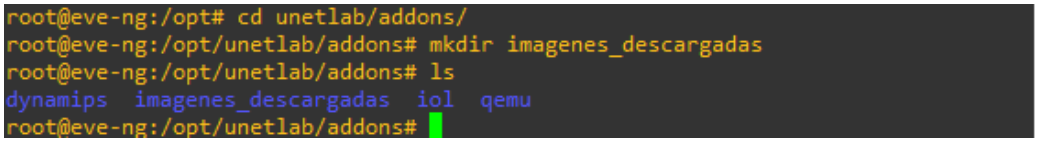


Figura 86. Carga de imágenes en Dynamips parte 1
Fuente: Elaboración propia

Transferir archivos al directorio creado mediante el programa Win SCP

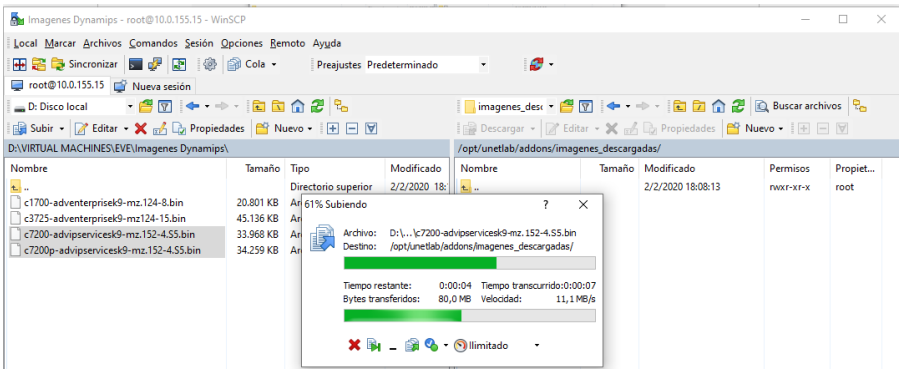


Figura 87. Carga de imágenes en Dynamips parte 2
Fuente: Elaboración propia

Descomprimir imágenes haciendo uso de la CLI, y cambiando la extensión de cada archivo a *.image*, como se muestra en la tabla antes indicada.

unzip -p c1710-bk9no3r2sy-mz.124-23.bin > c1710-bk9no3r2sy-mz.124-23.image

```
root@eve-ng:/opt/unetlab/addons# cd imagenes_descargadas/
root@eve-ng:/opt/unetlab/addons/imagenes_descargadas# ls
c1700-adventerprisek9-mz.124-8.bin  c7200-advipservicesk9-mz.152-4.S5.bin
c3725-adventerprisek9-mz124-15.bin  c7200p-advipservicesk9-mz.152-4.S5.bin
root@eve-ng:/opt/unetlab/addons/imagenes_descargadas# unzip
unzip     unzipsfx
root@eve-ng:/opt/unetlab/addons/imagenes_descargadas# unzip -p c1700-adventerpri
sek9-mz.124-8.bin > c1700-adventerprisek9-mz.124-8.image
```

Figura 88. Carga de imágenes en Dynamips parte 3
Fuente: Elaboración propia

De forma gráfica con la herramienta WinSCP se puede comprobar la descompresión de imágenes

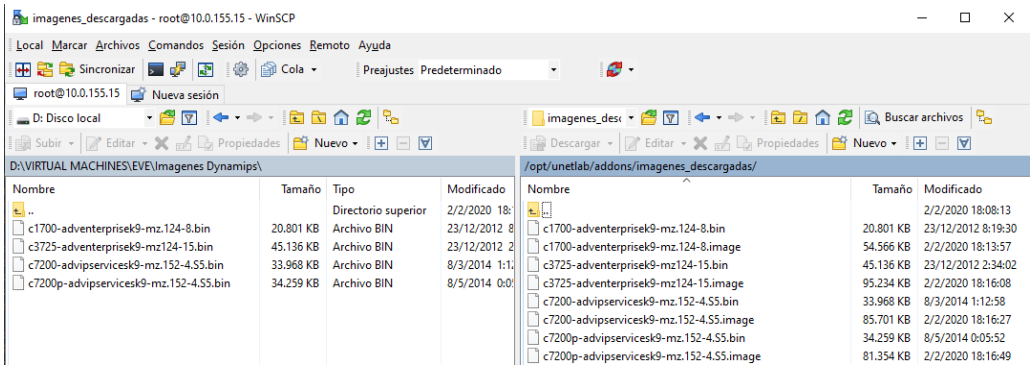


Figura 89. Carga de imágenes en Dynamips parte 4
Fuente: Elaboración propia

Mover las imágenes descomprimidas al directorio de Dynamips

mv c1710-bk9no3r2sy-mz.124-23.image /opt/unetlab/addons/dynamips/

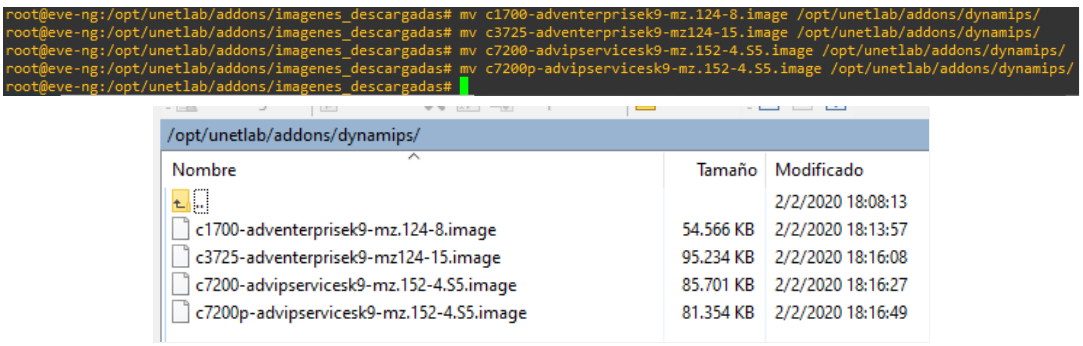
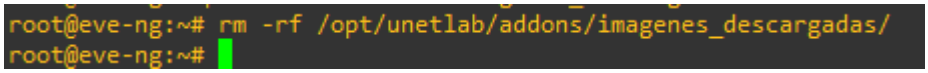


Figura 90. Carga de imágenes en Dynamips parte 5
Fuente: Elaboración propia

Borrar y dar permisos, se puede eliminar la carpeta que sirvió para subir las imágenes y a los archivos descomprimidos se les da permisos para poder trabajar con ellos

rm -rf /opt/unetlab/addons/imagenes_descargadas/



/opt/unetlab/wrappers/unl_wrapper -a fixpermissions

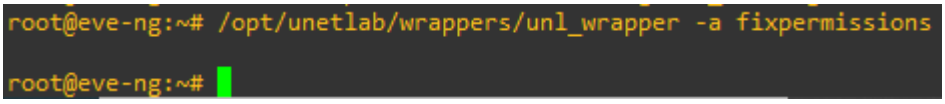


Figura 91. Carga de imágenes en Dynamips parte 6
Fuente: Elaboración propia

Calcular el valor de Idle-PC, este proceso permite identificar el mejor valor de Idle-PC, que permita el menor consumo de CPU, lo cual ayuda al rendimiento del hardware y de la red a simular. Para realiza este cálculo se puede ejecutar la imagen desde una línea de comandos de EVE, se toma como ejemplo la siguiente imagen:

dynamips -P 3725 /opt/unetlab/addons/dynamips/c3725-adventerprisek9-mz.124-15.T14.image

Se sale de la configuración de diálogo inicial hasta obtener el prompt

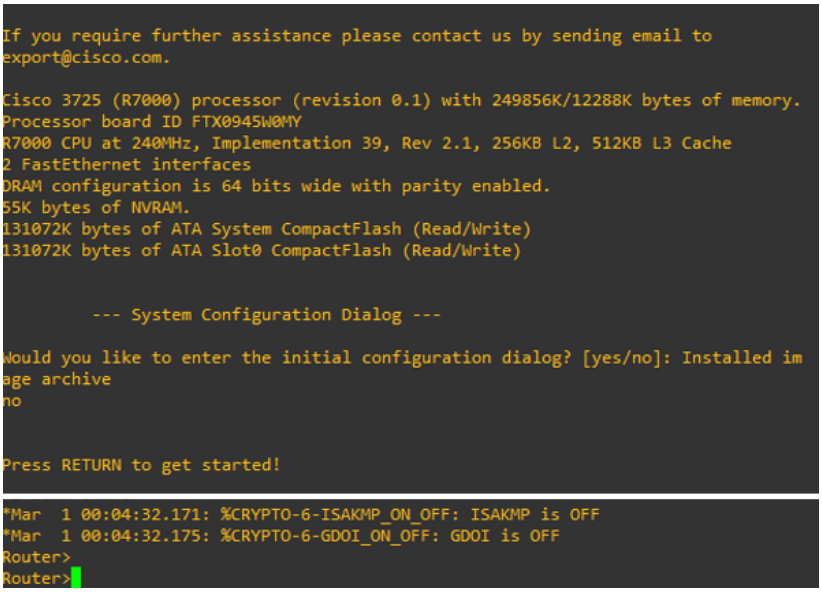


Figura 92. Carga de imágenes en Dynamips parte 7
Fuente: Elaboración propia

En otra ventana de línea de comandos, se va a observar el consumo de CPU mediante el comando **top**, que en primera instancia tiene alto porcentaje de consumo

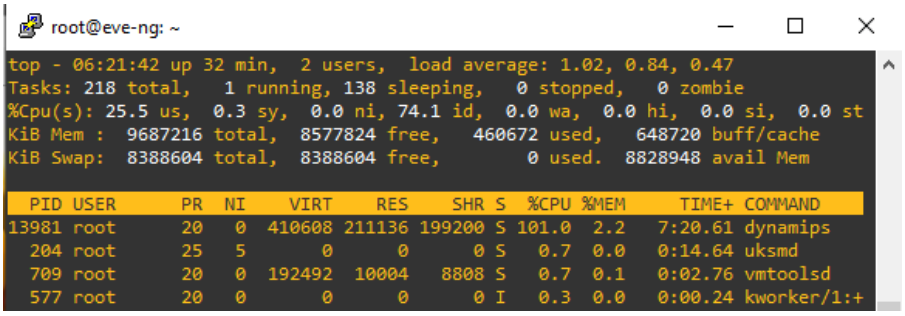


Figura 93. Carga de imágenes en Dynamips parte 8
Fuente: Elaboración propia

En la ventana inicial se presionan las teclas **Ctrl+]** y luego la tecla "**i**", para visualizar los diferentes valores de Idle-PC; por lo general el valor más alto es el recomendado para que el consumo de CPU sea lo más bajo posible, caso contrario se debe seguir probando con los otros valores.

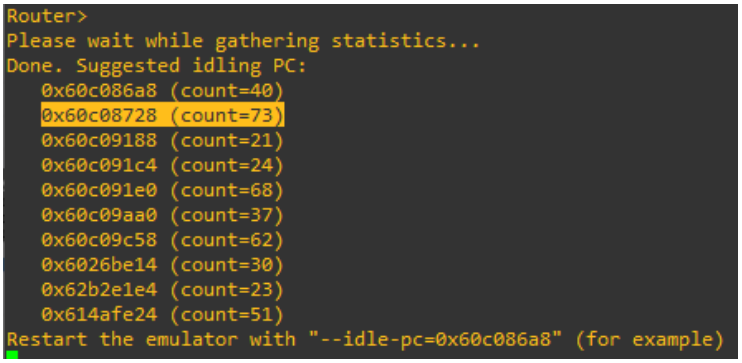


Figura 94. Carga de imágenes en Dynamips parte 9
Fuente: Elaboración propia

Para salir de esta pantalla se presionan las teclas **Ctrl+]** y luego la tecla "**q**". Se puede volver a ejecutar la imagen modificando el valor de Idle-PC desde la ventana de línea de comandos, o desde la GUI WEB de EVE. Se utiliza el método de comandos, y se observa si el consumo de CPU ha disminuido

dynamips -P 3725 --idle-pc=0x60c08728 /opt/unetlab/addons/dynamips/c3725-adventerprisek9-mz.124-15.T14.image

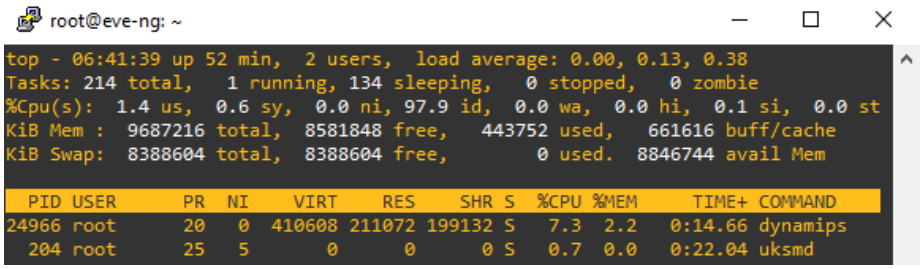


Figura 95. Carga de imágenes en Dynamips parte 10
Fuente: Elaboración propia

Este valor puede ser utilizado en los nodos a ejecutar desde la GUI WEB de EVE.

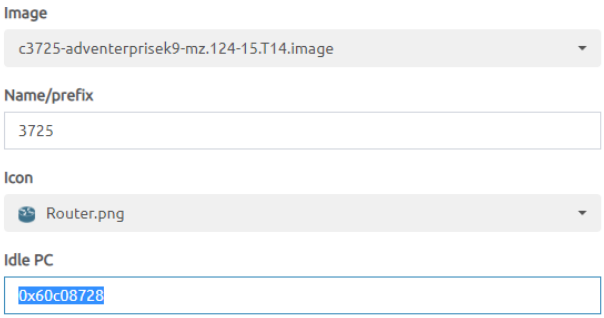


Figura 96. Carga de imágenes en Dynamips parte 11
Fuente: Elaboración propia

El procedimiento antes descrito es una recomendación de EVE, y que puede encontrarse en mayor detalle en la dirección https://www.eve-ng.net/index.php/documentation/howtos/howto-add-cisco-dynamips-images-cisco-ios/

7.1.4 CARGAR IMÁGENES EN QUEMU

Al igual que Dynamips, Quemu es un motor que permite emular imágenes de routers Cisco de arquitectura XR y que demandan mayor capacidad de CPU y RAM de un computador. Las imágenes soportadas son las que se muestran en el siguiente cuadro:

Tabla 8. Imágenes soportadas en Quemu

Imagen EVE	Archivo de descarga	Versión	vCPUs	vRAM
xrv-k9-5.1.1	iosxrv-k9-demo-5.1.1_2.ova	5.1.1	2	4096
xrv-k9-5.2.1	iosxrv-k9-demo-5.2.0.ova	5.2.1	1	3072
xrv-k9-5.2.2	iosxrv-k9-demo-5.2.2.ova	5.2.2	1	3072

Fuente: (EVE-NG, 2019)

- **DESCARGA DE IMÁGENES** Las imágenes de arquitectura XR de Cisco pueden ser descargadas de los siguientes enlaces

  https://mega.nz/#F!GIZ0lYYD!_Dlelfmvv8Dx6AVyhPi3gg!eBozxCqT

  https://upw.io/w5?pt=TjJob1MzWTNNV2hxVEVkWWEyVkhkRGRvVldKR1FUMDlPaEgvN3IzZmxkVmVGWlZOS3dTVzkvbz0%3D

- **CARGA DE IMÁGENES A EVE-NG** Para las versiones descritas y otras, el proceso es similar al utilizado para Dynamips:

  Crear un directorio temporal en una ubicación fácil de localizar, mediante el comando mkdir

  Cargar imagen a la carpeta creada y desde la línea de comandos descomprimir con tar xvf iosxrv-k9-demo-5.1.1_2.ova

  Convertir formato del archivo descomprimido a formato qcow2 con el comando /opt/qemu/bin/qemu-img convert -f vmdk -O qcow2 iosxrv-demo.vmdk hda.qcow2

  Crear carpetas por cada imagen y mover la imagen en formato qcow2 con los siguientes comandos o puede realizarse de forma gráfica desde WinSCP

  mkdir /opt/unetlab/addons/qemu/xrv-k9-5.1.1

  mv hda.qcow2 /opt/unetlab/addons/qemu/xrv-k9-5.1.1

  Borrar y dar permisos, se elimina la carpeta temporal y se generan permisos para poder utilizar las imágenes, con los siguientes comandos:

  cd ..

  rm -rf abc

  /opt/unetlab/wrappers/unl_wrapper -a fixpermissions

Figura 97. Carga de imágenes en Quemu parte 1
Fuente: Elaboración propia

En la GUI WEB de EVE, se puede ejecutar una o varias imágenes de arquitectura XR, con la observación de tener en cuenta el consumo de CPU y RAM, de acuerdo a las capacidades de hardware del equipo.

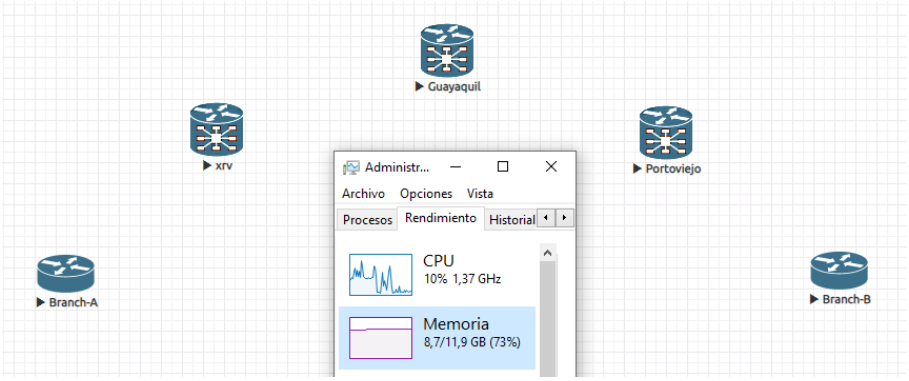


Figura 98. Carga de imágenes en Quemu parte 2
Fuente: Elaboración propia

Este procedimiento ha sido basado del sitio oficial de EVE que puede consultarse para mayor detalle en https://www.eve-ng.net/index.php/documentation/howtos/howto-add-cisco-xrv/

7.2 ANEXO 2 – MANEJO DE HERRAMIENTAS iPERF / jPERF

7.2.1 DESCRIPCIÓN DE iperf / jperf

iPerf es una herramienta que permite realizar medidas de ancho de banda en una red con modificaciones de parámetros relacionados a tiempo, buffers y protocolos (TCP, UDP, SCTP con IPv4/IPv6). Los resultados de cada prueba que se muestran son ancho de banda, packet loss y otros. iPerf3 es desarrollado por ESnet/Lawrence Berkeley National Laboratory. Sus características son las siguientes:

- TCP y SCTP
    - Medida de ancho de banda.
    - Reporte de MSS/MTU size.
    - Soporte para tamaño de ventana TCP por buffers socket.
- UDP
    - El cliente puede crear flujos de ancho de banda específico.
    - Medida de packet loss, delay jitter, capacidad multicast.
- Varias plataformas: Windows, Linux, Android, MacOs X, FreeBSD, OpenBSD, NetBSD, VxWorks, Solaris, entre otros.
- Se puede ejecutar conexión simultánea cliente – servidor (opción *–p*).
- El servidor puede manejar múltiples conexiones.
- Se puede ejecutar para un tiempo específico (opción *–t*), así como una cantidad de datos a transferir (opción *–n, -k*).
- Imprime reportes periódicos, intermedios de ancho de banda, jitter y packet loss (opción *–i*).
- Puede ejecutarse como daemon (opción *–D*).

- Usa streams representativos para probar cómo la compresión de la capa de enlace afecta al ancho de banda disponible (opción *–F*) (iPerf.fr, 2020).

jPerf es otra herramienta muy útil para monitoreo de estado de una red, con la diferencia que tiene una interface gráfica que se ejecuta como un archivo *.java*, de tal manera que en tiempo real el usuario puede observar el comportamiento de la red que está evaluando. (Behfor, 2018)

7.2.2 DESCARGA DE HERRAMIENTAS iPERF / jPERF

Las herramientas iperf, jperf se las puede descargar de forma gratuita desde Internet en los siguientes enlaces:

iperf https://iperf.fr/

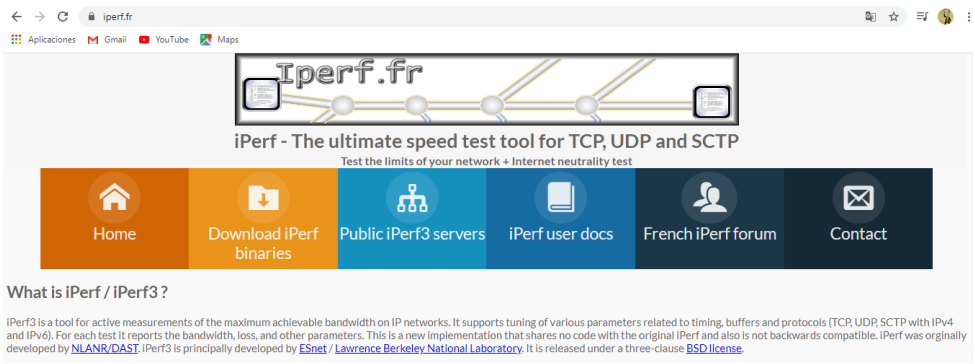


Figura 99. Página web de iperf
Obtenido de (iPerf.fr, 2020)

jperf https://code.google.com/archive/p/xjperf/downloads

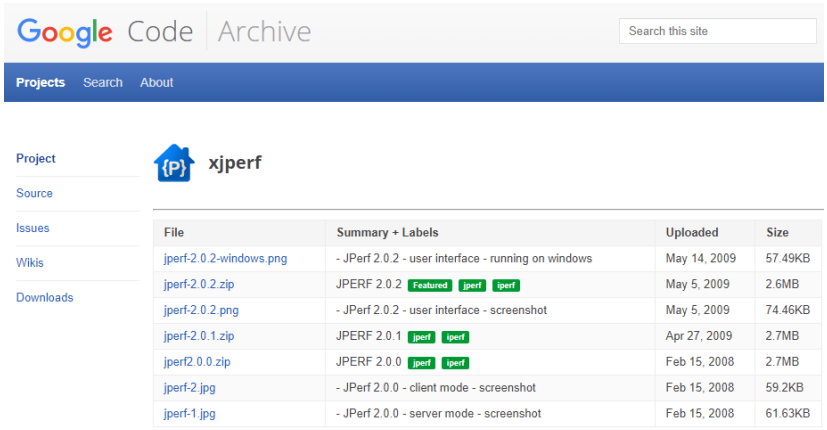


Figura 100. Página web de jperf
Obtenido de (Google Code Archive, s.f.)

No necesitan instalarse, y se ejecutan de la siguiente manera

- **EJECUCIÓN DE iperf,** después de descargar el archivo el mismo puede ser ubicado en un directorio de fácil acceso, mismo que después es llamado desde una ventana de línea de comandos tanto para servidor y cliente.

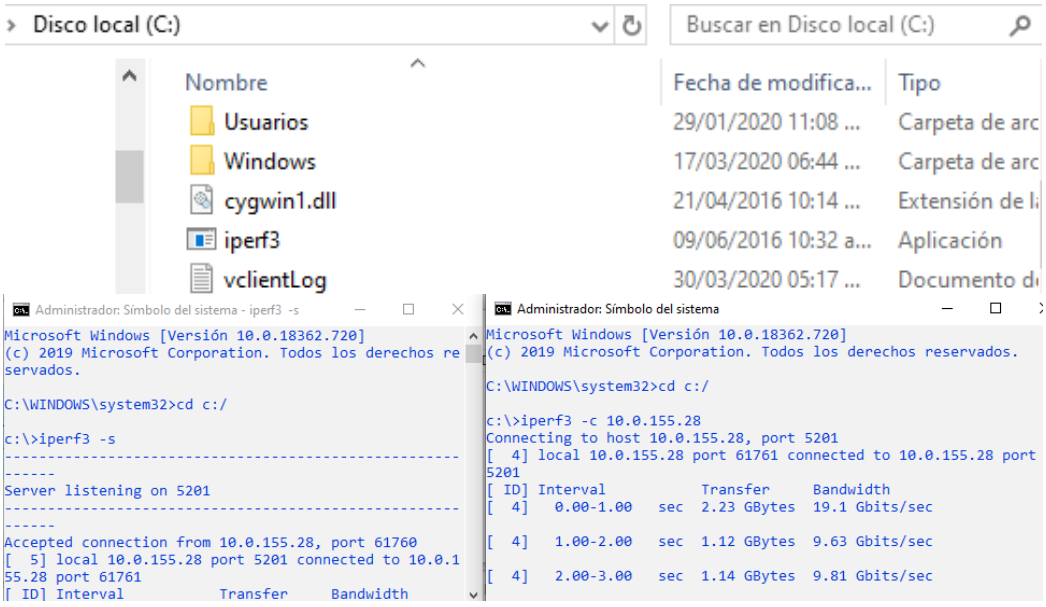


Figura 101. Ejecución de iperf
Fuente: Elaboración propia

- **EJECUCIÓN DE jperf,** una vez descargada esta herramienta, es necesario ejecutar el archivo .bat para acceder a la ventana gráfica como se muestra a continuación.

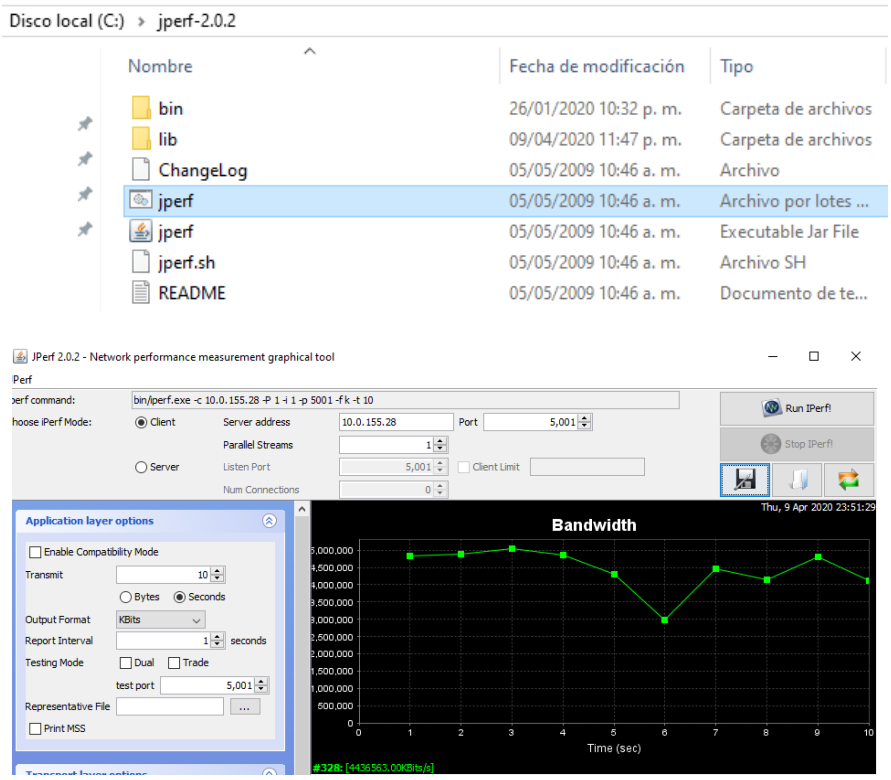


Figura 102. Ejecución de jperf
Fuente: Elaboración propia

7.3 ANEXO 3 – CONFIGURACIONES DE EQUIPOS MPLS L3VPN

En este anexo se detalla las configuraciones de los equipos simulados para el escenario de la arquitectura MPLS L3VPN que se presenta en la Figura 54.

7.3.1 EQUIPOS PE

PE1

```
RP/0/0/CPU0:PE1#show running-config

Sun Aug  9 04:46:51.229 UTC
Building configuration...
!! IOS XR Configuration 6.0.1
!! Last configuration change at Sun May 24 12:56:06 2020 by cisco
!
hostname PE1
telnet vrf default ipv4 server max-servers 10
telnet vrf default ipv6 server max-servers 10
username cisco
 secret 5 $1$Qpi0$sGA4X2zIxNO5Ht25ceX.u1
!
username benguele
 secret 5 $1$U0sJ$q7YqXshtmeje7fZCeI/Qr1
!
vrf netdat001
 address-family ipv4 unicast
  import route-target
   65000:1
  !
  export route-target
   65000:1
  !
 !
!
interface Loopback0
 ipv4 address 10.1.1.1 255.255.255.255
 ipv6 address 2001:db8:10:1:1::1/128
 ipv6 enable
!
interface MgmtEth0/0/CPU0/0
 shutdown
!
interface GigabitEthernet0/0/0/0
 description ###LINK-TO_CE1_Fa0/0###
 vrf netdat001
 ipv4 address 192.168.1.1 255.255.255.252
 ipv6 address 2001:db8:192:168:1::1/126
 ipv6 enable
!
interface GigabitEthernet0/0/0/1
 description ###LINK_TO_P01_Gi0/0/0/1###
 ipv4 address 192.168.12.2 255.255.255.252
 ipv6 address 2001:db8:192:168:12::2/126
 ipv6 enable
```

```
!
interface GigabitEthernet0/0/0/2
 shutdown
!
 route-policy PERMITE_TODO
  pass
end-policy
!
router isis 1
 net 49.0001.0100.0100.1001.00
 address-family ipv4 unicast
 !
 address-family ipv6 unicast
  single-topology
 !
 interface Loopback0
  address-family ipv4 unicast
  !
  address-family ipv6 unicast
  !
 !
 interface GigabitEthernet0/0/0/1
  circuit-type level-2-only
  address-family ipv4 unicast
  !
  address-family ipv6 unicast
  !
 !
!
router bgp 65000
 address-family vpnv4 unicast
 !
 neighbor 10.2.1.1
  remote-as 65000
  update-source Loopback0
  address-family vpnv4 unicast
   next-hop-self
  !
 !
 vrf netdat001
  rd 65000:1
  address-family ipv4 unicast
  !
  neighbor 192.168.1.2
   remote-as 65001
   update-source GigabitEthernet0/0/0/0
   address-family ipv4 unicast
    route-policy PERMITE_TODO in
    route-policy PERMITE_TODO out
   !
  !
 !
!
mpls ldp
 router-id 10.1.1.1
 interface GigabitEthernet0/0/0/1
 !
!
mpls label range table 0 16000 16099
end
```

PE2

```
RP/0/0/CPU0:PE2#show running-config

Sun Aug  9 04:51:54.209 UTC
Building configuration...
!! IOS XR Configuration 6.0.1
!! Last configuration change at Sun May 24 12:57:24 2020 by cisco
!
hostname PE2
telnet vrf default ipv4 server max-servers 10
telnet vrf default ipv6 server max-servers 10
username benguele
 secret 5 $1$04kH$/i.3GJymRdRLQqUT2voWH/
!
vrf netdat001
 address-family ipv4 unicast
  import route-target
   65000:1
  !
  export route-target
   65000:1
  !
 !
!
interface Loopback0
 ipv4 address 10.2.1.1 255.255.255.255
 ipv6 address 2001:db8:10:2:1::1/128
 ipv6 enable
!
interface MgmtEth0/0/CPU0/0
 shutdown
!
interface GigabitEthernet0/0/0/0
 description ###LINK_TO_CE2_Fa0/0###
 vrf netdat001
 ipv4 address 192.168.2.1 255.255.255.252
 ipv6 address 2001:db8:192:168:2::1/126
 ipv6 enable
!
interface GigabitEthernet0/0/0/1
 description ###LINK_TO_P01_Gi0/0/0/2###
 ipv4 address 192.168.21.2 255.255.255.252
 ipv6 address 2001:db8:192:168:21::2/126
 ipv6 enable
!
interface GigabitEthernet0/0/0/2
 shutdown
!
route-policy PERMITE_TODO
  pass
end-policy
!
router isis 1
 net 49.0002.0100.0200.1001.00
 address-family ipv4 unicast
 !
 address-family ipv6 unicast
  single-topology
 !
```

```
 interface Loopback0
  address-family ipv4 unicast
  !
  address-family ipv6 unicast
  !
 !
 interface GigabitEthernet0/0/0/1
  circuit-type level-2-only
  address-family ipv4 unicast
  !
  address-family ipv6 unicast
  !
 !
!
router bgp 65000
 address-family vpnv4 unicast
 !
 neighbor 10.1.1.1
  remote-as 65000
  update-source Loopback0
  address-family vpnv4 unicast
   next-hop-self
  !
 !
 vrf netdat001
  rd 65000:1
  address-family ipv4 unicast
  !
  neighbor 192.168.2.2
   remote-as 65002
   update-source GigabitEthernet0/0/0/0
   address-family ipv4 unicast
    route-policy PERMITE_TODO in
    route-policy PERMITE_TODO out
   !
  !
 !
!
 mpls ldp
 router-id 10.2.1.1
 interface GigabitEthernet0/0/0/1
 !
!
mpls label range table 0 16100 16199
end
```

7.3.2 **EQUIPOS P**

P01

```
RP/0/0/CPU0:P01#show running-config

Sun Aug  9 04:56:32.880 UTC
Building configuration...
!! IOS XR Configuration 6.1.3
!! Last configuration change at Sun May 24 13:00:10 2020 by cisco
!
hostname P01
```

```
telnet vrf default ipv4 server max-servers 10
telnet vrf default ipv6 server max-servers 10
username cisco
 secret 5 $1$Nykx$32V2gGVYhWkApWXIRaqrk.
!
username benguele
 secret 5 $1$SSK4$VobzwucZzDVO59r.VlZpu/
!
interface Loopback0
 ipv4 address 10.0.1.1 255.255.255.255
 ipv6 address 2001:db8:10:0:1::1/128
 ipv6 enable
!
interface MgmtEth0/0/CPU0/0
 shutdown
!
interface GigabitEthernet0/0/0/0
 description ###LINK_TO_P02_Gi0/0/0/0###
 ipv4 address 192.168.30.1 255.255.255.252
 ipv6 address 2001:db8:192:168:30::1/126
 ipv6 enable
!
interface GigabitEthernet0/0/0/1
 description ###LINK_TO_PE1_Gi0/0/0/1#
 ipv4 address 192.168.12.1 255.255.255.252
 ipv6 address 2001:db8:192:168:12::1/126
 ipv6 enable
!
interface GigabitEthernet0/0/0/2
 shutdown
!
router isis 1
 net 49.0000.0100.0000.1001.00
 address-family ipv4 unicast
 !
 address-family ipv6 unicast
  single-topology
 !
 interface Loopback0
  address-family ipv4 unicast
  !
  address-family ipv6 unicast
!
 !
 interface GigabitEthernet0/0/0/0
  circuit-type level-2-only
  address-family ipv4 unicast
  !
  address-family ipv6 unicast
  !
 !
 interface GigabitEthernet0/0/0/1
  circuit-type level-2-only
  address-family ipv4 unicast
  !
  address-family ipv6 unicast
  !
 !
!
mpls ldp
```

```
 router-id 10.0.1.1
 interface GigabitEthernet0/0/0/0
 !
 interface GigabitEthernet0/0/0/1
 !
!
mpls label range table 0 20000 20099
end
```

P02

```
RP/0/0/CPU0:P02#show running-config

Sun Aug  9 05:00:46.832 UTC
Building configuration...
!! IOS XR Configuration 6.1.3
!! Last configuration change at Sun May 24 13:01:02 2020 by cisco
!
hostname P02
telnet vrf default ipv4 server max-servers 10
username cisco
 secret 5 $1$Am66$SrmTEFag1le9YWJFoCtWD/
!
interface Loopback0
 ipv4 address 10.0.2.1 255.255.255.255
 ipv6 address 2001:db8:10:0:2::1/128
 ipv6 enable
!
interface MgmtEth0/0/CPU0/0
 shutdown
!
interface GigabitEthernet0/0/0/0
 description ###LINK_TO_P01_Gi0/0/0/0###
 ipv4 address 192.168.30.2 255.255.255.252
 ipv6 address 2001:db8:192:168:30::2/126
 ipv6 enable
!
interface GigabitEthernet0/0/0/1
 description ###LINK_TO_PE2_Gi0/0/0/1#
 ipv4 address 192.168.21.1 255.255.255.252
 ipv6 address 2001:db8:192:168:21::1/126
 ipv6 enable
!
interface GigabitEthernet0/0/0/2
 shutdown
!
router isis 1
 net 49.0000.0100.000.2001.00
 address-family ipv4 unicast
 !
 address-family ipv6 unicast
  single-topology
 !
 interface Loopback0
  address-family ipv4 unicast
  !
  address-family ipv6 unicast
  !
 !
```

```
 interface GigabitEthernet0/0/0/0
  circuit-type level-2-only
 address-family ipv4 unicast
  !
  address-family ipv6 unicast
  !
 !
 interface GigabitEthernet0/0/0/1
  circuit-type level-2-only
  address-family ipv4 unicast
  !
  address-family ipv6 unicast
  !
 !
!
mpls ldp
 router-id 10.0.2.1
 interface GigabitEthernet0/0/0/0
 !
 interface GigabitEthernet0/0/0/1
 !
!
mpls label range table 0 20100 20199
end
```

7.3.3 EQUIPOS CE

CE1

```
CE1#show running-config
Building configuration...

Current configuration : 1350 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname CE1
!
boot-start-marker
boot-end-marker
!
no aaa new-model
memory-size iomem 5
ip cef
!
ipv6 unicast-routing
!
multilink bundle-name authenticated
!
username cisco privilege 15 secret 5 $1$OwcW$u4N/aqzNWXWnYa6ZpQrta/
 archive
 log config
  hidekeys
!
interface Loopback0
```

```
 ip address 10.1.10.1 255.255.255.255
 ipv6 address 2001:DB8:10:1:10::1/128
 ipv6 enable
!
interface FastEthernet0/0
 description ###LINK_TO_PE1_Gi0/0/0/0/0###
 ip address 192.168.1.2 255.255.255.252
 duplex auto
 speed auto
 ipv6 address 2001:DB8:192:168:1::2/126
 ipv6 enable
 !
interface FastEthernet0/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
router bgp 65001
 no synchronization
 bgp log-neighbor-changes
 network 10.1.10.1 mask 255.255.255.255
 network 192.168.1.0 mask 255.255.255.252
 neighbor 192.168.1.1 remote-as 65000
 no auto-summary
!
ip forward-protocol nd
!
ip http server
no ip http secure-server
!
control-plane
!
line con 0
 exec-timeout 0 0
 privilege level 15
 logging synchronous
 login local
line vty 0 4
 logging synchronous
 login local
!
end
```

CE2

```
CE2#show running-config
Building configuration...

Current configuration : 1348 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname CE2
!
boot-start-marker
boot-end-marker
```

```
!
no aaa new-model
memory-size iomem 5
ip cef
!
ipv6 unicast-routing
!
 multilink bundle-name authenticated
!
username cisco privilege 15 secret 5 $1$bs4S$00TZ5bEg8fS2mqzkL8d3A0
 archive
 log config
  hidekeys
!
interface Loopback0
 ip address 10.2.10.1 255.255.255.255
 ipv6 address 2001:DB8:10:2:10::1/128
 ipv6 enable
!
interface FastEthernet0/0
 description ###LINK_TO_PE2_Gi0/0/0/0###
 ip address 192.168.2.2 255.255.255.252
 duplex auto
 speed auto
 ipv6 address 2001:DB8:192:168:2::2/126
 ipv6 enable
 !
interface FastEthernet0/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
router bgp 65002
 no synchronization
 bgp log-neighbor-changes
 network 10.2.10.1 mask 255.255.255.255
 network 192.168.2.0 mask 255.255.255.252
 neighbor 192.168.2.1 remote-as 65000
 no auto-summary
!
ip forward-protocol nd
!
ip http server
no ip http secure-server
!
control-plane
!
line con 0
 exec-timeout 0 0
 privilege level 15
 logging synchronous
 login local
line vty 0 4
 logging synchronous
 login local
!
end
```

7.4 ANEXO 4 – CONFIGURACIONES DE EQUIPOS SR MPLS (IPv4)

En este anexo se detalla las configuraciones de los equipos simulados para el escenario de la arquitectura SR MPLS que se presenta en la Figura 62.

7.4.1 EQUIPOS PE

PE1

```
RP/0/0/CPU0:PE1#sh run
Thu Oct 29 04:30:22.892 UTC
Building configuration...
!! IOS XR Configuration 6.0.1
!! Last configuration change at Sun Sep 20 14:22:43 2020 by cisco
!
hostname PE1
telnet vrf default ipv4 server max-servers 10
telnet vrf default ipv6 server max-servers 10
username puce
 secret 5 $1$/3M3$Ts6WfbYQZKlMmts9CwdPq0
!
username cisco
 secret 5 $1$Qpi0$sGA4X2zIxNO5Ht25ceX.u1
!
vrf netdat001
 address-family ipv4 unicast
  import route-target
   65000:1
  !
  export route-target
   65000:1
  !
 !
!
interface Loopback0
 ipv4 address 10.1.1.1 255.255.255.255
 ipv6 address 2001:db8:10:1:1::1/128
 ipv6 enable
!
interface MgmtEth0/0/CPU0/0
 shutdown
!
interface GigabitEthernet0/0/0/0
 description ###LINK-TO_CE1_Fa0/0###
 vrf netdat001
 ipv4 address 192.168.1.1 255.255.255.252
 ipv6 address 2001:db8:192:168:1::1/126
 ipv6 enable
!
interface GigabitEthernet0/0/0/1
 description ###LINK_TO_P01_Gi0/0/0/1###
 ipv4 address 192.168.12.2 255.255.255.252
 ipv6 address 2001:db8:192:168:12::2/126
 ipv6 enable
!
interface GigabitEthernet0/0/0/2
 shutdown
```

```
!
route-policy PERMITE_TODO
  pass
end-policy
!
router isis 1
 net 49.0001.0100.0100.1001.00
 address-family ipv4 unicast
  metric-style wide
  segment-routing mpls
 !
 address-family ipv6 unicast
  metric-style wide
  single-topology
  segment-routing mpls
 !
 interface Loopback0
  address-family ipv4 unicast
   prefix-sid index 0
  !
  address-family ipv6 unicast
  !
 !
 interface GigabitEthernet0/0/0/1
  circuit-type level-2-only
  address-family ipv4 unicast
  !
  address-family ipv6 unicast
  !
 !
!
router bgp 65000
 address-family vpnv4 unicast
 !
 neighbor 10.2.1.1
  remote-as 65000
  update-source Loopback0
  address-family vpnv4 unicast
   next-hop-self
  !
 !
 vrf netdat001
  rd 65000:1
  address-family ipv4 unicast
  !
  neighbor 192.168.1.2
   remote-as 65001
   update-source GigabitEthernet0/0/0/0
   address-family ipv4 unicast
    route-policy PERMITE_TODO in
    route-policy PERMITE_TODO out
   !
  !
 !
!
end
```

PE2

```
RP/0/0/CPU0:PE2#show running-config
Thu Oct 29 04:37:28.293 UTC
Building configuration...
!! IOS XR Configuration 6.0.1
!! Last configuration change at Tue Aug 18 11:08:21 2020 by cisco
!
hostname PE2
telnet vrf default ipv4 server max-servers 10
telnet vrf default ipv6 server max-servers 10
vrf netdat001
 address-family ipv4 unicast
  import route-target
   65000:1
  !
  export route-target
   65000:1
  !
 !
!
interface Loopback0
 ipv4 address 10.2.1.1 255.255.255.255
 ipv6 address 2001:db8:10:2:1::1/128
 ipv6 enable
!
interface MgmtEth0/0/CPU0/0
 shutdown
!
interface GigabitEthernet0/0/0/0
 description ###LINK_TO_CE2_Fa0/0###
 vrf netdat001
 ipv4 address 192.168.2.1 255.255.255.252
 ipv6 address 2001:db8:192:168:2::1/126
 ipv6 enable
!
interface GigabitEthernet0/0/0/1
 description ###LINK_TO_P01_Gi0/0/0/2###
 ipv4 address 192.168.21.2 255.255.255.252
 ipv6 address 2001:db8:192:168:21::2/126
 ipv6 enable
!
interface GigabitEthernet0/0/0/2
 shutdown
!
route-policy PERMITE_TODO
  pass
end-policy
!
router isis 1
 net 49.0002.0100.0200.1001.00
 address-family ipv4 unicast
  metric-style wide
  segment-routing mpls
 !
 address-family ipv6 unicast
  single-topology
 !
 interface Loopback0
  address-family ipv4 unicast
```

```
  prefix-sid index 1000
 !
 address-family ipv6 unicast
 !
!
interface GigabitEthernet0/0/0/1
 circuit-type level-2-only
 address-family ipv4 unicast
 !
 address-family ipv6 unicast
 !
!
!
router bgp 65000
 address-family vpnv4 unicast
 !
 neighbor 10.1.1.1
  remote-as 65000
  update-source Loopback0
  address-family vpnv4 unicast
   next-hop-self
  !
 !
 vrf netdat001
  rd 65000:1
  address-family ipv4 unicast
  !
  neighbor 192.168.2.2
   remote-as 65002
   update-source GigabitEthernet0/0/0/0
   address-family ipv4 unicast
    route-policy PERMITE_TODO in
    route-policy PERMITE_TODO out
   !
  !
 !
!
end
```

7.4.2 **EQUIPOS P**

P01

```
RP/0/0/CPU0:P01#show running-config
Thu Oct 29 04:43:48.607 UTC
Building configuration...
!! IOS XR Configuration 6.1.3
!! Last configuration change at Tue Aug 18 11:12:00 2020 by cisco
!
hostname P01
interface Loopback0
 ipv4 address 10.0.1.1 255.255.255.255
 ipv6 address 2001:db8:10:0:1::1/128
 ipv6 enable
!
interface MgmtEth0/0/CPU0/0
 shutdown
!
```

```
interface GigabitEthernet0/0/0/0
 description ###LINK_TO_P02_Gi0/0/0/0###
 ipv4 address 192.168.30.1 255.255.255.252
 ipv6 address 2001:db8:192:168:30::1/126
 ipv6 enable
!
interface GigabitEthernet0/0/0/1
 description ###LINK_TO_PE1_Gi0/0/0/1#
 ipv4 address 192.168.12.1 255.255.255.252
 ipv6 address 2001:db8:192:168:12::1/126
 ipv6 enable
!
interface GigabitEthernet0/0/0/2
 shutdown
!
router isis 1
 net 49.0000.0100.0000.1001.00
 address-family ipv4 unicast
  metric-style wide
  segment-routing mpls
 !
 address-family ipv6 unicast
  single-topology
 !
 interface Loopback0
  address-family ipv4 unicast
   prefix-sid index 4000
  !
  address-family ipv6 unicast
  !
 !
 interface GigabitEthernet0/0/0/0
  circuit-type level-2-only
  address-family ipv4 unicast
  !
  address-family ipv6 unicast
  !
 !
 interface GigabitEthernet0/0/0/1
  circuit-type level-2-only
  address-family ipv4 unicast
  !
  address-family ipv6 unicast
  !
 !
!
end
```

P02

```
RP/0/0/CPU0:P02#show running-config
Thu Oct 29 04:46:37.876 UTC
Building configuration...
!! IOS XR Configuration 6.1.3
!! Last configuration change at Tue Aug 18 11:15:58 2020 by cisco
!
hostname P02
interface Loopback0
```

```
 ipv4 address 10.0.2.1 255.255.255.255
 ipv6 address 2001:db8:10:0:2::1/128
 ipv6 enable
!
interface MgmtEth0/0/CPU0/0
 shutdown
!
interface GigabitEthernet0/0/0/0
 description ###LINK_TO_P01_Gi0/0/0/0###
 ipv4 address 192.168.30.2 255.255.255.252
 ipv6 address 2001:db8:192:168:30::2/126
 ipv6 enable
!
interface GigabitEthernet0/0/0/1
 description ###LINK_TO_PE2_Gi0/0/0/1#
 ipv4 address 192.168.21.1 255.255.255.252
 ipv6 address 2001:db8:192:168:21::1/126
 ipv6 enable
!
interface GigabitEthernet0/0/0/2
 shutdown
!
router isis 1
 net 49.0000.0100.000.2001.00
 address-family ipv4 unicast
  metric-style wide
  segment-routing mpls
 !
 address-family ipv6 unicast
  single-topology
 !
 interface Loopback0
  address-family ipv4 unicast
   prefix-sid index 5000
  !
  address-family ipv6 unicast
  !
 !
 interface GigabitEthernet0/0/0/0
  circuit-type level-2-only
  address-family ipv4 unicast
  !
  address-family ipv6 unicast
  !
 !
 interface GigabitEthernet0/0/0/1
  circuit-type level-2-only
  address-family ipv4 unicast
  !
  address-family ipv6 unicast
  !
 !
!
End
```

7.4.3 **EQUIPOS CE**

CE1

```
CE1#show running-config
Building configuration...

Current configuration : 1350 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname CE1
!
boot-start-marker
boot-end-marker
!
no aaa new-model
memory-size iomem 5
ip cef
!
ipv6 unicast-routing
!
multilink bundle-name authenticated
!
username cisco privilege 15 secret 5 $1$OwcW$u4N/aqzNWXWnYa6ZpQrta/
archive
 log config
  hidekeys
!
interface Loopback0
 ip address 10.1.10.1 255.255.255.255
 ipv6 address 2001:DB8:10:1:10::1/128
 ipv6 enable
!
interface FastEthernet0/0
 description ###LINK_TO_PE1_Gi0/0/0/0/0###
 ip address 192.168.1.2 255.255.255.252
 duplex auto
 speed auto
 ipv6 address 2001:DB8:192:168:1::2/126
 ipv6 enable
!
interface FastEthernet0/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
router bgp 65001
 no synchronization
 bgp log-neighbor-changes
 network 10.1.10.1 mask 255.255.255.255
 network 192.168.1.0 mask 255.255.255.252
 neighbor 192.168.1.1 remote-as 65000
 no auto-summary
!
ip forward-protocol nd
```

```
!
ip http server
no ip http secure-server
!
control-plane
!
line con 0
 exec-timeout 0 0
 privilege level 15
 logging synchronous
 login local
line aux 0
 exec-timeout 0 0
 privilege level 15
 logging synchronous
line vty 0 4
 logging synchronous
 login local
!
!
end
```

CE2

```
CE2#show running-config
Building configuration...

Current configuration : 1348 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname CE2
!
boot-start-marker
boot-end-marker
!
no aaa new-model
memory-size iomem 5
ip cef
!
ipv6 unicast-routing
!
multilink bundle-name authenticated
!
username cisco privilege 15 secret 5 $1$bs4S$00TZ5bEg8fS2mqzkL8d3A0
archive
 log config
  hidekeys
!
interface Loopback0
 ip address 10.2.10.1 255.255.255.255
 ipv6 address 2001:DB8:10:2:10::1/128
 ipv6 enable
!
interface FastEthernet0/0
 description ###LINK_TO_PE2_Gi0/0/0/0###
```

```
 ip address 192.168.2.2 255.255.255.252
 duplex auto
 speed auto
 ipv6 address 2001:DB8:192:168:2::2/126
 ipv6 enable
!
interface FastEthernet0/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
router bgp 65002
 no synchronization
 bgp log-neighbor-changes
 network 10.2.10.1 mask 255.255.255.255
 network 192.168.2.0 mask 255.255.255.252
 neighbor 192.168.2.1 remote-as 65000
 no auto-summary
!
ip forward-protocol nd
!
ip http server
no ip http secure-server
!
control-plane
!
line con 0
 exec-timeout 0 0
 privilege level 15
 logging synchronous
 login local
line aux 0
 exec-timeout 0 0
 privilege level 15
 logging synchronous
line vty 0 4
 logging synchronous
 login local
!
end
```

7.5 ANEXO 5 – CONFIGURACIONES DE EQUIPOS SRv6

En este anexo se detalla las configuraciones de los equipos simulados para el escenario de la arquitectura L3VPN basada en SRv6 que se presenta en la Figura 72.

7.5.1 EQUIPOS PE

PE1

```
RP/0/0/CPU0:PE1#sh run

Thu Dec 17 04:05:50.827 UTC
Building configuration...
!! IOS XR Configuration 6.6.1
!! Last configuration change at Wed Dec 16 05:47:39 2020 by cisco
!
hostname PE1
!
segment-routing srv6
 locators
  locator myLoc1
  prefix 2001:db8:a1:1::/64
 !
!
vrf netdat001
 address-family ipv4 unicast
  import route-target
   65000:1
  !
  export route-target
   65000:1
  !
 !
!
interface Loopback0
 ipv6 address 2001:db8:10:1:1::1/128
 ipv6 enable
!
interface MgmtEth0/0/CPU0/0
 shutdown
!
interface GigabitEthernet0/0/0/0
 description ###LINK_TO_CE1_Fa0/0###
 ipv4 address 192.168.1.1 255.255.255.252
!
interface GigabitEthernet0/0/0/1
 description ###LINK_TO_P01_Gi0/0/0/1###
 ipv6 address 2001:db8:192:168:12::2/126
 ipv6 enable
!
interface GigabitEthernet0/0/0/2
 shutdown
!
route-policy PERMITE_TODO
  pass
end-policy
!
```

```
router isis 1
 net 49.001.0100.0100.1001.00
 address-family ipv6 unicast
  single-topology
  segment-routing srv6
   locator myLoc1
  !
 !
 interface Loopback0
  address-family ipv6 unicast
  !
 !
 interface GigabitEthernet0/0/0/1
  circuit-type level-2-only
  address-family ipv6 unicast
  !
 !
!
router bgp 65000
 bgp router-id 10.1.1.1
 address-family ipv4 unicast
 !
 address-family vpnv4 unicast
  segment-routing srv6
   locator myLoc1
  !
 !
 neighbor 10.2.1.1
  remote-as 65000
  update-source Loopback0
  address-family vpnv4 unicast
   next-hop-self
  !
 !
 vrf netdat001
  rd 65000:1
  address-family ipv4 unicast
   segment-routing srv6
    alloc mode per-vrf
   !
  !
  neighbor 192.168.1.2
   remote-as 65001
   update-source GigabitEthernet0/0/0/0
   address-family ipv4 unicast
    route-policy PERMITE_TODO in
    route-policy PERMITE_TODO out
   !
  !
 !
!
end
```

PE2

```
RP/0/0/CPU0:PE2#sh run

Thu Dec 17 04:06:49.804 UTC
Building configuration...
```

```
!! IOS XR Configuration 6.6.1
!! Last configuration change at Wed Dec 16 05:48:48 2020 by cisco
!
hostname PE2
!
segment-routing srv6
 locators
  locator myLoc1
  prefix 2001:db8:a1:2::/64
!
 !
vrf netdat001
 address-family ipv4 unicast
  import route-target
   65000:1
  !
  export route-target
   65000:1
  !
 !
!
interface Loopback0
 ipv6 address 2001:db8:10:2:1::1/128
 ipv6 enable
!
interface MgmtEth0/0/CPU0/0
 shutdown
!
interface GigabitEthernet0/0/0/0
 description ###LINK_TO_CE2_Fa0/0###
 ipv4 address 192.168.2.1 255.255.255.252
!
interface GigabitEthernet0/0/0/1
 description ###LINK_TO_P02_Gi0/0/0/1###
 ipv6 address 2001:db8:192:168:21::2/126
 ipv6 enable
!
interface GigabitEthernet0/0/0/2
 shutdown
!
route-policy PERMITE_TODO
  pass
end-policy
!
router isis 1
 net 49.0002.0100.0200.1001.00
 address-family ipv6 unicast
  single-topology
   segment-routing srv6
   locator myLoc1
  !
 !
 !
 interface Loopback0
  address-family ipv6 unicast
  !
 !
 interface GigabitEthernet0/0/0/1
  circuit-type level-2-only
  address-family ipv6 unicast
```

```
 !
 !
!
router bgp 65000
 bgp router-id 10.2.1.1
 address-family ipv4 unicast
 !
 address-family vpnv4 unicast
  segment-routing srv6
   locator myLoc1
  !
 !
 neighbor 10.1.1.1
  remote-as 65000
  update-source Loopback0
  address-family vpnv4 unicast
   next-hop-self
  !
 !
 vrf netdat001
  rd 65000:1
  address-family ipv4 unicast
   segment-routing srv6
    alloc mode per-vrf
   !
  !
 neighbor 192.168.2.2
   remote-as 65002
   update-source GigabitEthernet0/0/0/0
   address-family ipv4 unicast
    route-policy PERMITE_TODO in
    route-policy PERMITE_TODO out
   !
  !
 !
!
end
```

7.5.2 **EQUIPOS P**

P01

```
RP/0/0/CPU0:P01#sh run

Thu Dec 17 04:07:39.010 UTC
Building configuration...
!! IOS XR Configuration 6.6.1
!! Last configuration change at Wed Dec 16 04:24:20 2020 by cisco
!
hostname P01
interface Loopback0
 ipv6 address 2001:db8:10:0:1::1/128
 ipv6 enable
!
interface MgmtEth0/0/CPU0/0
 shutdown
```

```
!
interface GigabitEthernet0/0/0/0
 description ###LINK_TO_P02_Gi0/0/0/0###
 ipv6 address 2001:db8:192:168:30::1/126
 ipv6 enable
!
interface GigabitEthernet0/0/0/1
 description ###LINK_TO_PE1_Gi0/0/0/1###
 ipv6 address 2001:db8:192:168:12::1/126
 ipv6 enable
!
interface GigabitEthernet0/0/0/2
 shutdown
!
router isis 1
 net 49.0000.0100.0000.1001.00
 address-family ipv6 unicast
  single-topology
 !
 interface Loopback0
  address-family ipv6 unicast
  !
 !
 interface GigabitEthernet0/0/0/0
  circuit-type level-2-only
  address-family ipv6 unicast
  !
 !
 interface GigabitEthernet0/0/0/1
  circuit-type level-2-only
  address-family ipv6 unicast
  !
 !
!
end
```

P02

```
RP/0/0/CPU0:P02#sh run

Thu Dec 17 04:08:06.738 UTC
Building configuration...
!! IOS XR Configuration 6.6.1
!! Last configuration change at Wed Dec 16 04:31:28 2020 by cisco
!
hostname P02
interface Loopback0
 ipv6 address 2001:db8:10:0:2::1/128
 ipv6 enable
!
interface MgmtEth0/0/CPU0/0
 shutdown
!
interface GigabitEthernet0/0/0/0
 description ###LINK_TO_P01_Gi0/0/0/0###
 ipv6 address 2001:db8:192:168:30::2/126
 ipv6 enable
!
```

```
interface GigabitEthernet0/0/0/1
 description ###LINK_TO_PE2_Gi0/0/0/1###
 ipv6 address 2001:db8:192:168:21::1/126
 ipv6 enable
!
interface GigabitEthernet0/0/0/2
 shutdown
!
router isis 1
 net 49.0000.0100.0000.2001.00
 address-family ipv6 unicast
  single-topology
 !
 interface Loopback0
  address-family ipv6 unicast
  !
 !
 interface GigabitEthernet0/0/0/0
  circuit-type level-2-only
  address-family ipv6 unicast
  !
 !
 interface GigabitEthernet0/0/0/1
  circuit-type level-2-only
  address-family ipv6 unicast
  !
 !
!
end
```

7.5.3 **EQUIPOS CE**

CE1

```
CE1#sh running-config
Building configuration...

Current configuration : 1022 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname CE1
!
boot-start-marker
boot-end-marker
!
no aaa new-model
memory-size iomem 5
ip cef
!
multilink bundle-name authenticated
!
archive
 log config
hidekeys
!
```

```
interface Loopback0
 ip address 10.1.10.1 255.255.255.255
!
interface FastEthernet0/0
 description ###LINK_TO_PE1_Gi0/0/0/0###
 ip address 192.168.1.2 255.255.255.252
 duplex auto
 speed auto
!
interface FastEthernet0/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
router bgp 65001
 no synchronization
 bgp router-id 10.1.10.1
 bgp log-neighbor-changes
 network 10.1.10.1 mask 255.255.255.255
 network 192.168.1.0 mask 255.255.255.252
 neighbor 192.168.1.1 remote-as 65000
 no auto-summary
!
ip forward-protocol nd
!
ip http server
no ip http secure-server
!
control-plane
!
line con 0
line aux 0
line vty 0 4
 login
!
end
```

CE2

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

Current configuration : 997 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname CE2
!
boot-start-marker
boot-end-marker
!
no aaa new-model
memory-size iomem 5
ip cef
!
```

```
multilink bundle-name authenticated

!
archive
 log config
 hidekeys

!
interface Loopback0
 ip address 10.2.10.1 255.255.255.255
!
interface FastEthernet0/0
 description ###LINK_TO_PE2_Gi0/0/0/0###
 ip address 192.168.2.2 255.255.255.252
 duplex auto
 speed auto
!
interface FastEthernet0/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
router bgp 65002
 no synchronization
 bgp log-neighbor-changes
 network 10.2.10.1 mask 255.255.255.255
 network 192.168.2.0 mask 255.255.255.252
 neighbor 192.168.2.1 remote-as 65000
 no auto-summary
!
ip forward-protocol nd
!
ip http server
no ip http secure-server
!
control-plane
!
line con 0
line aux 0
line vty 0 4
 login
!
end
```