ESCUELA POLITÉCNICA NACIONAL

ESCUELA DE FORMACIÓN DE TECNÓLOGOS

VIRTUALIZACIÓN DE UNA RED MPLS CON VPNS EN UNA *CLOUD COMPUTING*

TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO DE TECNÓLOGO SUPERIOR EN REDES Y TELECOMUNICACIONES

ANA PAULA PANCHES MORA

DIRECTOR: ING. FERNANDO VINICIO BECERRA CAMACHO

DMQ, Febrero 2022

CERTIFICACIONES

Yo, Ana Paula Panches Mora declaró que el trabajo de integración curricular aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

Ana Paula Panches Mora

anapaula.panches@epn.edu.ec

anapaulapanches327@gmail.com

Certifico que el presente trabajo de integración curricular fue desarrollado por Ana Paula Panches Mora, bajo mi supervisión.

Fernando Becerra

fernando.becerrac@epn.edu.ec

DECLARACIÓN DE AUTORÍA

A través de la presente declaración, afirmamos que el trabajo de integración curricular aquí descrito, así como el producto resultante del mismo, son públicos y estarán a disposición de la comunidad a través del repositorio institucional de la Escuela Politécnica Nacional; sin embargo, la titularidad de los derechos patrimoniales nos corresponde a los autores que hemos contribuido en el desarrollo del presente trabajo; observando para el efecto las disposiciones establecidas por el órgano competente en propiedad intelectual, la normativa interna y demás normas.

ANA PAULA PANCHES MORA

DEDICATORIA

Dedico mi trabajo de titulación a mis padres por todo el sacrificio y esfuerzo para que yo tenga una buena educación y poder tener una carrera profesional, por formarme con buenos valores y hábitos. Lo que me permitió salir adelante con cada obstáculo en el camino, por acompañarme en cada paso que di y siempre estar orgullosos de mí.

A mis hermanos que fueron los que siempre me sacaban una sonrisa cuando ya sentía que no podía, por tener su apoyo y ayudarme en lo que lograban entender. Por siempre estar unidos y preocuparnos por el bienestar de todos nosotros.

A mis abuelos por siempre apoyarme y creer en mí, decirme que lo iba a lograr. Por estar el día que me gradúe del colegio. Por felicitarme cuando entre a la universidad y ayudarme a ser la persona que soy hoy en día.

AGRADECIMIENTO

Agradecimiento al ING. Fernando Becerra por asesorarme y apoyarme durante todo el proceso del trabajo de titulación; un excelente profesor durante este tiempo que he pasado en la universidad, sin su ayuda y conocimiento no podría haber realizo con éxito este proyecto.

Agradecimiento a mis padres que siempre estuvieron pendiente de mi proyecto, que siempre buscaron la forma de ayudarme o preguntarme cómo me está yendo al realizar mi trabajo de titulación. Que creyeron en mí durante este proceso que, aunque no comprendía del todo de qué trataba, estuvieron escuchándome.

ÍNDICE DE CONTENIDOS

RESUMEN

Para la implementación de cuatro redes utilizando VPNs (*Virtual Private Network*) y MPLS (*Multiprotocol Label Switching*) en *cloud computing*. Se realizaron varias investigaciones, para conocer los diferentes tipos de VPNs sobre MPLS, al igual que los equipos de *networking* que soporten la configuración de las mismas.

En el capítulo 1 se tendrá una introducción del proyecto implementado y los objetivos a seguir para realizar las redes de VPN de capas 2 y 3. Al igual que el alcance que tendrá el proyecto.

En el capítulo 2 se investigará sobre las diferentes *cloud computing* que existen en el mercado, al igual que los simuladores de red que se puedan implementar sobre una *cloud computing*. Al igual que la investigación sobre los equipos de *networking* que soporten MPLS y VPNs. Todo esto para evitar inconvenientes al momento de configurar los *routers* y perder tiempo en esta tarea.

Siguiendo con el capítulo 2 se investigará los tipos de VPNs sobre MPLS, a partir de la investigación se comenzará con la implementación del EVE-NG (*Emulated Virtual Environment)* sobre Google Cloud. Al igual que la configuración de las VPNS sobre MPLS, donde fue necesario ocupar tanto los protocolos de enrutamiento dinámico como estático.

En el capítulo 3 se realizó las pruebas de funcionamiento, donde se ocupó comandos como *ping* o *traceroute q*ue permitió probar la conexión de extremo a extremo y conocer el camino que toma el paquete. También se observó las tablas de enrutamiento de MPLS y de los protocolos de enrutamiento.

PALABRAS CLAVE: *cloud computing*, simulador de red, ubuntu, VPNs, MPLS.

ABSTRACT

For the implementation of four networks using VPN (Virtual Private Network) and MPLS (Multiprotocol Label Switching) in cloud computing. Several investigations were carried out to learn about the different types of VPN over MPLS, as well as the networking equipment that supports their configuration.

Chapter 1 there will be an introduction of the implemented project and the objectives to follow to carry out the VPN networks of layers 2 and 3. As well as the scope of the project.

Chapter 2, the different cloud computing that exist in the market will be investigated, as well as the network simulators that can be implemented on cloud computing. As well as research on networking equipment that supports MPLS and VPN. All this to avoid inconveniences when configuring both routers.

Continuing with chapter 2, the types of VPNs over MPLS will be investigated, starting with the investigation, the implementation of EVE-NG (Emulated Virtual Environment) over Google Cloud will begin. Like the configuration of VPNS over MPLS, where it was necessary to use both dynamic and static routing protocols.

Chapter 3, the performance tests were carried out, where commands such as ping or traceroute were used, which showed us the end-to-end connection and knowing the path that the packet takes. Also note the MPLS routing tables and routing protocols.

KEYWORDS: cloud computing, network simulator, ubuntu, VPNs, MPLS.

1 DESCRIPCIÓN DEL COMPONENTE DESARROLLADO

Se utilizó un emulador que entre las varias opciones que se investigó se decidió por EVE-NG, este emulador va a ser montado en una *cloud computing* por el consumo de recursos de cómputo. Para la elección de *cloud computing* se decidió por *Google Cloud*, ya que esta solución dispone de una versión gratuita la que se encuentra disponible por tres meses o 300 dólares de consumo.

Para la creación de las diferentes redes de datos se utilizaron equipos de *networking* los cuales son de la marca Cisco, después se realizó el direccionamiento IP y sobre esta se agregó protocolos de enrutamiento (RIPv2 (*Router Information Protocol Version 2*), EIGRP (*Enhanced Interior Gateway Routing Protocol*), OSPF (*Open Shortest Path First*) y BGP (*Border Gateway Protocol*)). Estos protocolos fueron implementados y además se tomó un tiempo hasta que estos se converjan, después se procedió a configurar MPLS en cada *router* que se encuentran en las redes diseñadas.

En el caso de MPLS puede utilizar cualquiera de los protocolos antes mencionados, ya que su configuración no cambia dependiendo del protocolo de enrutamiento sólo se necesita una conectividad entre los vecinos. Al ya tener MPLS en funcionamiento se comienza con la configuración de dos VPNs una de capa 2 y otra de capa 3.

Para el caso de las VPNs de capa 3 se debe implementar el protocolo de enrutamiento BGP y redistribuir la ruta estática entre los equipos que vayan a ocupar las VPNs, para que todos los enrutadores conozcan esa ruta y pueda existir conexión. En el caso de las VPNs de capa 2 no necesitan de un protocolo en específico para su configuración tanto para MPLS como las VPNs

Se implementaron un total de cuatro redes, dos por cada tipo de VPN sobre MPLS. Al terminar con la configuración de las redes se procedió a realizar las pruebas de funcionamiento como también se observaron las tablas que se generan tanto por los protocolos de enrutamiento como de MPLS para ver si efectivamente se generaron las VPNs.

1.1 Objetivo general

Implementación y virtualización de redes MPLS con VPNs a partir de un simulador de red sobre una *cloud computing*.

1.2 Objetivos específicos

1. Investigar simuladores de red que se puedan implementar sobre una *cloud computing.*

2. Investigar sobre los equipos de *networking* que soporten MPLS y VPNs.

3. Investigar los tipos de VPNs sobre MPLS.

4. Implementar un simulador de red en una *cloud computing.*

5. Implementar las VPNs sobre MPLS en una red sobre *cloud computing.*

6. Realizar pruebas de funcionamiento.

1.3 Alcance

El presente proyecto tiene como alcance la implementación de al menos dos redes MPLS mediante un simulador de red utilizando VPNs en una *cloud computing*. Se investigará por lo menos tres simuladores de red que permitan la configuración de equipos *networking* que soporten tanto MPLS como VPNs. Se investigará al menos tres *cloud computing* que permitan implementar máquinas virtuales y que por otro lado soporten los simuladores de red, de las tres opciones se seleccionará la mejor opción. Se realizará un estudio de los diferentes tipos de VPNs que se pueden utilizar sobre MPLS.

Al obtener la información sobre los distintos equipos de *networking, s*e procede a investigar varias ISOs para obtener la adecuada y colocarla en el simulador de red, todo esto para evitar problemas de configuración tanto de los protocolos de enrutamiento que son necesarios para configurar VPNs sobre MPLS, se realizará todo lo antes mencionado para tener resultados satisfactorios en las diferentes redes implementadas.

Al terminar con la configuración de las redes tanto de VPNs de capa 2 y 3. Se comienza con las pruebas de funcionamiento, para estas se ocuparán varios comandos que permitan visualizar si la configuración de red fue realizada de forma correcta. Además, se obtendrá tablas de enrutamiento de los distintos protocolos a ocupar en cada una de las redes, al igual que la tabla de MPLS para conocer el próximo salto, las etiquetas que se están ocupando entre otros parámetros. También se ocupará comandos como *ping* y *traceroute* que permitan observar si los paquetes llegan al destino y el camino que ocupa para llegar al mismo

1.4 Marco Teórico

SIMULADOR

Es una herramienta que permite representar el comportamiento de distintos equipos dentro del programa, los simuladores solo son utilizados para observar el tráfico, protocolos, gráficos y paquetes enviados en tiempo real. La simulación de una red es un método basado en modelos del comportamiento de una red, esta se puede simular mediante cálculos de interacción entre los distintos *hosts* o elementos de red, los enlaces de datos o por paquetes, a partir de la utilización de fórmulas matemáticas. También por captura de tráfico con pruebas que permitan conocer la conexión entre los equipos de la red [1].

El simulador de red es una pieza que es de software que permite predecir el comportamiento de una red mediante el manejo de equipos de *networking* virtuales, con este software no se tiene la necesidad de tener los equipos de hardware que son demasiado costosos. El simulador de red imita el funcionamiento de una red informática de datos, en esta herramienta permite modelar los equipos a ocupar en la red real. Normalmente estos programas llegan a soportar los protocolos más conocidos como lo son IP, Protocolos de enrutamiento, protocolos WAN, UDP, TCP,etc.. [1].

EMULACIÓN

Es un sistema que imita un proceso real, donde va a ocupar recursos de un sistema de cómputo para crear entornos virtuales o múltiples plataformas independientes para ejecutar programas propios. Cada uno de ellos va a usar propios modelos, dando como resultado que el sistema de hardware original no perciba una única aplicación [2].

CLOUD COMPUTING

Cuenta con soluciones dependiendo de las necesidades informáticas primarias: *infrastructure as a service* (IaaS), *platform as a service* (PaaS) y *software as a service* (SaaS) [3].

- IaaS: permite la ejecución de máquinas virtuales sin la necesidad de invertir o administrar una infraestructura informática. Cuando se opta por esta solución es porque la carga de trabajo es temporal o experimental o tiende a tener cambios inesperados [3].

- PaaS: los clientes deciden optar por PaaS por todos los beneficios que presenta, ya que obtiene una infraestructura subyacente, como sistemas operativos y *middleware.* El proveedor hospeda y gestiona todos estos elementos [3].
- SaaS: el proveedor va a administrar toda la infraestructura, incluida aplicaciones que ocupa el cliente. En el caso de los usuarios ellos solo deben iniciar sesión y podrán acceder a todos los recursos como lo es herramientas de respaldo o recuperación [3].

Multiprotocol Label Switching

Es una tecnología que unifica la transferencia de varios datos en una misma red, todo esto para superar las limitaciones que existe en la velocidad también tiene un aumento en el flujo de trabajo del Internet. Este sistema permite transmitir a través de una misma red de gran área geográfica diferentes tipos de datos, entre ellos están: VoIP, servicios de video vigilancia [4].

Las estaciones intermedias LSR (*Label Switched Router*) las que establecían la mejor ruta para el paquete de datos. En cambio, MPLS brinda la posibilidad de predeterminar las rutas que determinan el camino que va a seguir el paquete desde el punto de origen hasta el destino. Debido a esto MPLS logra descongestionar la gran carga de trabajo que los sistemas de enrutamiento soportaban en la antigüedad [5].

MPLS es utilizado esencialmente por operadores que quieren garantizar la calidad de servicio en gestión de tráfico, así como en VPN. Ahora, el funcionamiento de MPLS se realiza añadiendo en cada paquete de datos que va a ser enviado (estos pueden ser voz, texto, video) un encabezado. Estos abarcan una o varias etiquetas agrupadas dependiendo de la operación que se lleva a cabo en los enrutadores por donde el paquete va viajando hasta llegar a su destino [5].

LDP (*Label Distribution Protocol*) es un protocolo utilizado para distribuir etiquetas en aplicaciones que no cuenta con ingeniería de tráfico. LDP permite a los enrutadores fijar una ruta de acceso conmutada por las etiquetas LSP. Es ejecutado en equipos que son compatibles con MPLS, ya que establece automáticamente adyacencias de LSP [6].

VPN

Una VPN permite tener una conexión segura entre las redes, este sería el caso del Internet o en el de una red de un proveedor de Internet. Utilizada frecuentemente para interconectar dos lugares a partir de la infraestructura de una red, ya que es mucho más segura debido a que cuenta con cifrado para permitir garantizar una transmisión segura

de los datos. Ayudando a que el usuario pueda trabajar de manera remota y segura. Debido a esto son utilizadas ampliamente en los entornos empresariales [7].

2 METODOLOGÍA

Al iniciar el proyecto de titulación se investigó tres simuladores de red que se pueden implementar sobre una c*loud computing.* De las diferentes opciones que se encuentran en Internet y al comparar sus características se decidió utilizar el simulador EVE-NG. Para el caso de las c*loud computing* se tuvo cuatro opciones. De las cuatro opciones se realizó un cuadro comparativo entre los servicios que ofrece, el costo y el punto de vista del cliente. A partir de esto se escogió la plataforma de Google Cloud.

Además, se investigó varios equipos de *networking* y su respectiva imagen ISO que soporte MPLS como también las VPN tanto de capa dos como tres. No solo fue necesario que las imágenes contaran con esos protocolos ya mencionados, sino que soportara varios protocolos de enrutamiento dinámico como lo es OSPF y BGP. Todo esto debido a que en la capa de las VPNs de capa 3 solo pueden ser implementadas en una red con el protocolo BGP. A parte de estas características se debió conocer si la imagen se encontraba en el formato que permite EVE-NG, para cargar la imagen del equipo.

También fue necesario investigar la utilidad de las VPNs sobre MPLS, donde se obtuvo que son usadas para conexiones de tipo sitio a sitio. Después se indagó los tipos de VPNs sobre MPLS. Existen dos tipos las VPNs de capa 2 o conocidas como VPLS (*Virtual Private LAN Service*), el segundo tipo son las VPNS de capa 3 o también conocidas como VPRF (*Virtual Routing and Forwarding*).

Para la implementación de EVE-NG en Google Cloud. Fue necesario ocupar la imagen de Ubuntu 16.04, ya que en esta versión de Ubuntu se puede instalar la versión c*ommunity*. En la creación de la instancia fue necesario buscar una zona geográfica que permita tener cargas de trabajo de uso general como lo son las N1. Al tener creada la instancia, se utiliza el protocolo SSH para entrar en la máquina virtual e instalar EVE-NG para realizar las debidas configuraciones.

Se implementaron cuatro redes, dos por cada tipo de VPNs sobre MPLS. En el caso de las dos redes con VPN de capa 2, no se tuvo que ocupar protocolos de enrutamiento dinámico específicos tanto para la configuración de MPLS como de las VPNs. En el caso de las dos redes con VPN de capa 3, en la configuración de MPLS se realizó lo mismo

que en las VPNS de capa 2. Pero en el caso de la configuración de las VPNS se tuvo que ocupar el protocolo de enrutamiento BGP.

Para la realización de pruebas se ocupó tanto con el comando *ping c*omo con el t*raceroute*. Este último para conocer el camino que hace los paquetes para llegar a su destino. También se observó las tablas de enrutamiento al igual que la tabla de MPLS. Para conocer las etiquetas que se ocupa en cada r*outer* y la ruta por donde viaja el paquete que también ocupa las VPNs.

3 RESULTADOS Y DISCUSIÓN

En el presente trabajo de titulación se crearon cuatro redes, dos de VPN de capa 2 y las otras dos de VPN de capa 3. Para la configuración de las dos redes que van a utilizar VPN de capa 2, es necesario ocupar protocolos de enrutamiento tanto dinámico como estático, para la creación de las VPNs como para la configuración de MPLS.

En el caso de las redes de VPN de capa 3 se debe ocupar igual protocolos de enrutamiento, pero para la creación de las VPNs se debe ocupar el protocolo BGP, esto aumenta su complejidad en la implementación.

Para el caso de las cuatro redes es necesario ocupar interfaces de *loopback*, ya que estas son utilizas para los protocolos de enrutamiento, como de MPLS debido a que se ocupa como *router ID*. Al terminar con la configuración se realiza pruebas de funcionamiento al igual que se observa las tablas tanto de enrutamiento como de MPLS y se realiza envíos de paquetes utilizando el nombre de la VPN y la dirección IP de esta.

3.1 Investigar simuladores de red que se puedan implementar sobre una *cloud computing*

SIMULADORES DE RED

EVE-NG

EVE-NG se encuentra desplegado sobre un Ubuntu y contiene los paquetes necesarios para ejecutar la emulación de las redes y contiene los *scripts* que generan una interfaz web para poder utilizar Qemu[1] y Dynamips[2] [8].

[1] Qemu: es un s*oftware* que se reproduce en una máquina virtual. La máquina virtual contiene su propio procesador [18].
[2] Dynamips: es un emulador solo de equipos Cisco, que permite familiarizar con los comandos de los dispositivos Cisco [19].

Entre sus características se tiene las siguientes:

- Interfaz de usuario HTML5 [9].
- Capacidad de simulación sin herramientas adicionales [9].
- Interacción total con la red real [9].
- Cuenta con soporte en el servidor Ubuntu LTS 16.04 [9].

GNS3 (*Graphical Network Simulator-3*)

GNS3 es un software ocupado a nivel mundial, entre sus funciones están simular, configurar, probar y solventar problemas de redes tanto virtuales como reales. Permite ejecutar desde pequeñas redes que constan de pocos equipos en una computadora portátil. Este simulador tiene equipos virtuales que permiten realizar una red más compleja [10].

GNS3 simula las funcionalidades de un equipo como lo es un *router.* Este simulador no ejecuta sistemas operativos reales, como lo hace Cisco IOS, sino que utiliza un dispositivo simulado desarrollado por GNS3 [10].

PNETLab (*Packet Network Emulator Tool Lab*)

Es una plataforma que permite descargar y compartir laboratorios con la comunidad que existe en este simulador. Incluye PNETLab Box y PNETLab *store* [11].

- PNETLab Box: existen dos modos el *offline* y *online.* Se encuentra instalada en una máquina local y el laboratorio se ejecuta sobre la máquina virtual, debido a esto el usuario no se preocupa por la velocidad del laboratorio [11].
- PNETLab Store: es una plataforma web que cuenta con laboratorios gratuitos en campos como redes, bases de datos, entre otros [11].

Cuenta con las imágenes ISO de Cisco, Juniper, entre otros, ya que estos se encuentran incluidos al momento de descargarlo en la página oficial [11].

Entre los diferentes tipos de simuladores se decidió ocupar el simulador de EVE-NG, ya que existe mayor información sobre su instalación en Google Cloud, al igual que su manejo para la implementación de redes es más sencillo [11].

En la Tabla 3.1 se realizó una comparación de los tres simuladores de red investigados. Entre los parámetros que se comparó está su origen, el acceso a imágenes ISO, si cuenta con información de implementación de c*loud computing* y el terminal de usuario.

Al observar esta información el simulador más completo es el EVE-NG, que cuenta con manual de usuario para su implementación y el formato que necesita la imagen ISO para cargarse en el simulador.

Tabla 3.1 Tabla comparativa entre los diferentes simuladores [11] [12]

Simulador de red	Origen	Acceso a imágenes ISO	Implementado en una c*loud computing*	Condiciones de aplicación del terminal
EVE-NG	Cuenta con versión gratuita y profesional.	Se puede acceder a partir de un contrato de servicios.	Cuenta con un manual para su instalación en Google Cloud,	Se puede ocupar la interfaz HTML5 o ocupar un terminal liviano como Putty.
GNS3	Cuenta con una interfaz gratuita y de código abierto.	Solo permite acceder a partir de un contrato de servicios.	No se cuenta con mucha información para su implementación.	Requiere de una terminal para modificar y probar el funcionamiento
PNETLab	Es una plataforma gratuita, que permite crear y compartir con equipos de múltiples proveedores.	Cuenta con imágenes Cisco, Juniper, Arista entre otros.	Existe información de su implementación en la plataforma de Google Cloud al igual que EVE-NG.	La aplicación del terminal tiene por consola local o puede ocuparse HTML.

CLOUD COMPUTING

AMAZON WEB SERVICES

Amazon Web Services abarca una gran variedad de servicios para la realización de distintas actividades en la c*loud computing*. Cuenta con servicios de almacenamiento a la gestión de instancias, imágenes virtuales, desarrollo de aplicaciones, entre otros. La nube de Amazon ha logrado establecerse a lo largo de los años como una de las mejoras en el mercado [13].

Amazon EC2 permite la creación de entornos virtuales en la *cloud*. Esto se da a partir de una interfaz web que se conecta con una imagen de máquina de Amazon conocida como AMI (Amazon *Machine Image)*. La AMI suele pertenecer al sistema operativo que se quiere ejecutar en la máquina virtual o también conocida como instancia, todo esto depende de la terminología que se usa en Amazon EC2 [13].

GOOGLE CLOUD

Google Cloud aporta todos los instrumentos necesarios para diseñar, realizar pruebas y lanzar aplicaciones desde Google Cloud. Estas funciones tienen mayor seguridad y escalabilidad que cualquier otra herramienta, todo esto a partir de la infraestructura con la que cuenta Google [14].

Una de sus funciones es *Compute Engine* que cuenta con cargas de trabajo de uso general como lo son las E2, N1, N2 que son máquinas que ofrecen un buen equilibrio entre el precio y el rendimiento. Son ocupadas para una amplia variedad de cargas de trabajo comunes como lo son las bases de datos, entornos de desarrollo, realización de pruebas, aplicaciones y juegos para equipos móviles [14].

OPENSHIFT

Openshift es una plataforma que cuenta con características de *Cloud Computing* del tipo PaaS que es presentada y ofertada por la empresa Red Hat [15].

A partir de esta plataforma cualquier desarrollador podrá darle a su aplicación la versatilidad que necesita, ya que podrá ajustarse a las regulaciones de cualquier país, resguardar datos de manera segura, asegurar su funcionamiento y brindar un rendimiento más rápido y confiable [15]

Esta plataforma que trabaja tanto en nube pública como privada, cuenta con los beneficios de una nube híbrida. Consiguiendo un mayor control de las cargas de trabajo, una disminución de costos y una gran facilidad para administrar y gestionar todo lo que se relacione con la aplicación a desarrollar [15].

AZURE DE MICROSOFT

Azure es una plataforma que cuenta solo con una *cloud* pública de pago por uso que cuenta con varias herramientas entre ellas compilar, implementar y administrar aplicaciones en una red global de *datacenters* (centros de datos) de Microsoft [16].

En el portal de Azure existen distintos servicios de infraestructura y de plataforma que permite al cliente montar los servicios que llegue a necesitar de forma sencilla. Por ejemplo, se tiene el caso de crear una máquina virtual, donde se permite seleccionar el tipo de máquina como puede llegar a ser Windows Server 2016 *datacenter,* para después rellenar todas las características como lo es la RAM y el espacio del disco. Durante el proceso de creación, se define tanto el nombre de usuario como la contraseña creada por cada usuario para el inicio de sesión en la máquina virtual creada [16].

En la Tabla 3.2 se puede observar las comparaciones como lo son los servicios, costo y punto de vista del cliente de las *cloud computing,* se decidió utilizar la plataforma de Google Cloud, con el simulador de red EVE-NG que solo puede instalarse en Google Cloud. Una de las ventajas que tuvo Google Cloud sobre el resto de plataformas fue su costo, ya que da un saldo gratis por tres meses de 300 dólares.

Tabla 3.2 Tabla comparativa entre las diferentes c*loud computing* **[3]**.

Tipo de *Cloud Computing*	Servicios	Costo	Punto de vista en el cliente
Google Cloud	Cuenta con menos servicios, pero tiene capacidades técnicas avanzadas como lo es *Machine Learning.* Al igual que aplicaciones de análisis de datos.	Cuenta con un saldo gratis por tres meses de 300 dólares. También tiene grandes descuentes y contratos flexibles.	Puede ser ocupada por pequeñas y medianas empresas o de forma personal.
Amazon *Web Services*	Cuenta con varias opciones, pero estas se encuentran centradas en soluciones de nube pública. Todo eso llega a provocar problemas en una implementación híbrida.	Herramientas de forma gratuita. Aunque también se puede ocupar dispositivos de terceros para evaluar los precios de manera efectiva.	Puede ser ocupado tanto por empresas como de forma personal, pero con un enfoque de no intervención debido a su tamaño.
Azure de Microsoft	Cuenta con una infraestructura robusta, que también ha sido probada en la nube híbrida.	Existen ciertos descuentos, para empresas dependiendo del producto que se necesite.	Es solo utilizado por empresas.
Openshift	Puede trabajar en una nube pública como privada. También cuenta con beneficios al contar con una nube híbrida, ya que permite tener un mayor control de las cargas de trabajo, entre otras.	Cuenta con varios precios dependiendo de la memoria de la instancia, ya que su costo se da por hora.	Puede ser ocupado tanto por empresas como de forma personal.

3.2 Investigar sobre los equipos de *networking* que soporten MPLS y VPNs

Entre las funciones del software Cisco, se debe encontrar las siguientes:

- Funciones básicas de enrutamiento y conmutación.

- Tener un acceso confiable y seguro a los recursos de la red.
- Tener la opción de escalabilidad de la red.

Para el arranque del IOS cuenta con una secuencia predeterminada tanto para su ubicación como para cargar la imagen. Al momento de arrancar un dispositivo Cisco se realiza una comprobación del hardware, para después intentar cargar la imagen IOS desde una memoria o un servidor TFTP. En caso de no encontrar la imagen ejecutará la versión reducida de esa IOS ubicada en la memoria ROM.

Cisco IOS XRv

Ofrece una mayor agilidad, eficiencia de la red mejorada y la capacidad de escalar eficientemente la red, dependiendo de la demanda [17].

Entre sus características principales están:

- Extremadamente resistente, estable en funciones administrativas al igual que garantizar una integración fluida [17].
- Solución integral con infraestructura de virtualización de funciones de red virtual y administración de servicios [17].
- Cuanta con varios hipervisores compatibles entre ellos está VMware, Ubuntu 16.04 LTS al igual que CentOS [17].
- Entre los diferentes protocolos de enrutamiento cuenta con MPLS, LDP y VPN [17].
- Cuenta con funciones del plano de datos como la calidad de servicio, listado de control de acceso y el reenvío de rutas [17].

Cisco CSR 1000v

Este tipo de enrutador que promete funciones integrales de servicios de red y puertas de enlace WAN en los entornos virtuales al igual que en la *cloud.* Debido a esto el CSR 1000v permite a las empresas ampliar de forma transparente sus enlaces WAN a *clouds* alojadas por los proveedores [17].

Entre sus características principales están:

- Brinda seguridad y servicios de red de clase empresarial en entornos de la c*loud* pública [17].
- Puede utilizarse como un elemento básico para servicios de red escalables [17].

- Entre las funciones que puede desempeñar se encuentra las VPN, firewall, *Network Address Translation* (NAT), calidad de servicio y la optimización de WAN [17].
- También cuenta con autenticación y contabilidad al igual que protocolos de configuración dinámico de host [17].

  [17].

En la siguiente Tabla 3.3 se realizó una comparación entre las características que tiene estas dos de imágenes ISO, donde se pudo observar que la IOS XRv puede ser ocupado individualmente, ya que solo necesita la activación de las licencias. También cuenta con varios protocolos de enrutamiento entre ellos MPLS y VPN en comparación del CSR 1000v que solo cuenta con las VPNs. Entre sus hipervisores compatibles el XRv cuenta con Ubuntu 16.04 LTS lo cual es una ventaja ya que EVE-NG c*ommunity* está implementado en esta versión.

Tabla 3.3 Tabla comparativa entre las imágenes ISO de Cisco **[17]**.

Imagen ISO Cisco	Tipos de uso	Protocolos de enrutamiento	Hipervisores compatibles
Cisco IOS XRv	Puede ser utilizado tanto en el ámbito empresarial como individual, ya que solo se necesita activar las licencias.	Cuenta con varios protocolos de enrutamiento básicos, también se puede utilizar MPLS, LDP y VPN.	Tiene diferentes hipervisores entre ellos VMware, Ubuntu 16.04, entre otros.
Cisco CSR 1000v	Ocupado más en el ámbito empresarial, ya que cuenta con una mayor seguridad en la c*loud* pública.	Cuenta con varios protocolos de enrutamiento básicos, pero en este caso ya permite la configuración de VPN.	Puede ser implementado en la c*loud* y en Ubuntu.

3.3 Investigar los tipos de VPNs sobre MPS

VPN

Una VPN protege la conexión a Internet a cualquier ordenador para garantizar que todos los paquetes transmitidos estén codificados y ocultos de cualquier ciberdelincuente. Entre sus diferentes características se tiene:

- Permite tener mayor anonimato en el Internet, todo esto debido a que la dirección IP y la ubicación están ocultas, o se puede conectar a un servidor distinto que se puede encontrar en otra ciudad o país [7].
- Existe mayor libertad al navegar en el Internet, ya que al utilizar diferentes direcciones IP, podrá acceder a sitios web y servicios en línea que, de otro modo se encontrarían bloqueados [7].
- La seguridad aumenta al ocupar el Internet, debido al túnel cifrado ya que mantendrá alejados a los piratas informáticos y los ciberdelincuentes y el dispositivo no será tan vulnerable a los ataques [7].

Las VPNs sobre MPLS son utilizadas con mayor eficiencia para las conexiones del tipo sitio a sitio. Esto se da sobre todo por el hecho que la tecnología de MPLS es una de las opciones más manejables y flexibles. Se trata de un recurso de base estándar que es ocupado para incrementar la distribución de paquetes de una red con múltiples protocolos. Las VPNs sobre MPLS son sistemas que se encuentran estrechados con los ISP (*Internet Service Provider).* Se le conoce a una VPN ajustada a ISP cuando dos o más lugares se encuentran conectados para formar una VPN, usando el mismo ISP. Existe dos tipos de VPNs sobre MPLS, estas son las de capa dos y tres [18].

Capa 2 VPN

La Capa 2 de VPNs MPLS, o conocida como VPLS, brinda un tipo de servicio conocido como “conmutador en la nube”. Las VPLS proporcionan la capacidad de ampliar las VLANs (Virtual LAN) entre sitios. La VPNs de capa 2 son utilizadas típicamente para enrutar el tráfico que puede ser de voz, video y AMI entre las subestaciones y los d*atacenters* [19].

La VPN de capa 2 permite extender varias redes lógicas de su misma capa y logra llegar a los límites de la capa 3, que son las encargadas de hacer el túnel dentro de SSL VPN. Además, consigue configurar diferentes sitios en un servidor de VPLS. Se tiene como ejemplo que las VPN de capa 2 permite que las empresas migren sin problema alguno en sus cargas de trabajo, ya que estas están respaldadas por las VLANs que se encuentran en ubicaciones separadas físicamente. En el caso de los proveedores de nube, las VPLS proporcionan un mecanismo para incluir nuevas empresas sin tener que modificar las direcciones IP ya existentes tanto de las cargas de trabajo como de las aplicaciones [19].

Capa 3 VPN

La Capa 3 de VPNs MPLS o también conocida como VPRN (*Virtual Private Routed Network*), utiliza a la capa 3 con VRF para separar las tablas de enrutamiento para cada cliente que se encuentre utilizando el servicio. Para su funcionamiento el cliente compara la tabla del *router* del proveedor de servicios y las dos rutas de intercambio, que están colocadas en una tabla de enrutamiento específica para cada cliente. Una de las características al momento de configurar VPRN es la utilización del protocolo BGP, ya que es necesario en la nube para ocupar el servicio. Todo esto aumenta la complejidad del diseño y su implementación [7].

Las VPNs de capa 3 generalmente no son utilizadas en el despliegue de las redes de servicios público, ya que como se mencionó anteriormente es bastante complejo su diseño e implementación; sin embargo, podrían ser usadas para enrutar el tráfico entre diferentes ubicaciones corporativas o d*atacenters* [7].

3.4 Implementar un simulador de red en una *cloud computing*

El inconveniente que existió al momento de instalar EVE-NG *Community*, fue que en las máquinas virtuales actuales en Google Cloud que ejecutan la imagen de Ubuntu 16.04 LTS se encuentra marcada como obsoleta. Debido a esto no permite recibir actualizaciones de seguridad al momento de ejecutar proyectos con esta imagen.

Para solucionar este problema lo primero que se realizó fue utilizar el comando que se encuentra en la Figura 3.1, para obtener las distintas imagenes ISO con las que cuenta *Google Cloud* y a partir de esta buscar la imagen de Ubuntu 16.04 *minimal* que se podrá visualizar en la Figura 3.2

```
anapaulapanches327@cloudshell:~ (eveng1-332423)$ gcloud compute images list
```

Figura 3.1 Comando para listado de imágenes ISO de Google Cloud

```
NAME: ubuntu-minimal-1604-xenial-v20210430
PROJECT: ubuntu-os-cloud
FAMILY: ubuntu-minimal-1604-lts
DEPRECATED:
STATUS: READY
```

Figura 3.2 Imagen de Ubuntu 16.04 *minimal*

Para la creación del *base disk* se utilizó el comando que se encuentra en la Figura 3.3**,** en este comando se coloca la imagen de Ubuntu, con el nombre de su familia que está en la Figura 3.2 y para el caso de la zona geográfica es necesario conocer las regiones disponibles y cuantos tipos de máquina soporta cada una de ellas, para la instalación

del simulador de red se decidió ocupar la zona de Asia del Este, la decisión fue tomado debido a que existía problemas al momento de crear la instancia.

```
gcloud compute instances create temp-image-base –image-family=projects/ubuntu-os-cloud/global/images/family/ubuntu-minimal-1604-lts --zone=asia-east1-a
```

Figura 3.3 Comando para la creación del base disk

El comando que se utilizó se encuentra en la Figura 3.4, el cual permitirá detener la imagen que se está ejecutando actualmente, para que Google Cloud comprenda que se requiere una máquina virtual que está anidada a la imagen de Ubuntu.

```
gcloud compute instances stop temp-image-base —zone=asia-east1-a
```

Figura 3.4 Comando para detener la imagen de Ubuntu

El siguiente comando que se encuentra en la Figura 3.5**,** este comando va a permitir habilitar la virtualización anidada, se toma en cuenta que la zona que se ocupó en el comando anterior debe ser el mismo para este, después de esto se va a obtener una nueva imagen del disco creado con la habilitación de la virtualización anidada.

```
gcloud compute images create nested -vm-image –source-disk=temp-image-base source-disk-zone=asia-east1-a --licenses="https://www.googleapis.com/compute/v1/projects/vm-options/global/licenses/enable-vmx"
```

Figura 3.5 Comando que permite habilitar la virtualización

Se crea la máquina virtual anidada que tendrá cuatro núcleos y el tamaño del disco de arranque es de 100 GB. El comando a ocupar se encuentra en la Figura 3.6.

```
gcloud compute instances create nested-vm –zone asia-east1-a --image=nested-vm-image –machine-type=n1-standard-4 –boot-disk-side=100GB
```

Figura 3.6 Comando para creación de máquina virtual

Se podrá visualizar la nueva instancia anidada en VM *Instances* en la Figura 3.7, que ya se encuentra en funcionamiento por lo que ahora se conectará con SSH para la instalación de EVE-NG.

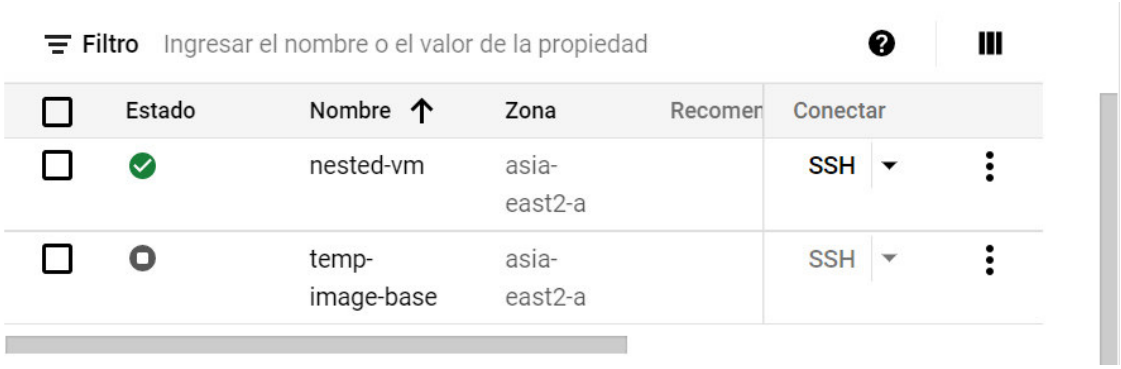

Figura 3.7 Creación de la nueva instancia

Al conectarnos con SSH se entra en Ubuntu 16.04, lo primero a realizar es tener acceso *root* con el comando en la Figura 3.8**,** en esta misma figura se tiene el comando que hace referencia al *script* de instalación del EVE-NG que se encuentra en el repositorio del mismo.

```
anapaulapanches327@nested-vm:~$ sudo -i
root@nested-vm:~# wget -O - http://www.eve-ng.net/repo/install-eve.sh | bash -i
```

Figura 3.8 Comando para tener acceso root

Después de la instalación de EVE-NG se debe reiniciar la máquina virtual, para realizar la configuración del simulador de red al terminar con esta configuración, la instancia se vuelve a reiniciar, pero para esta última ya no es necesario conectarse con el SSH, si no ocupar la IP pública que aparece en el panel de VM *Instances,* al colocar esta IP en el buscador se debe presentar la pantalla de inicio del EVE-NG, que se podrá observar en la Figura 3.9.

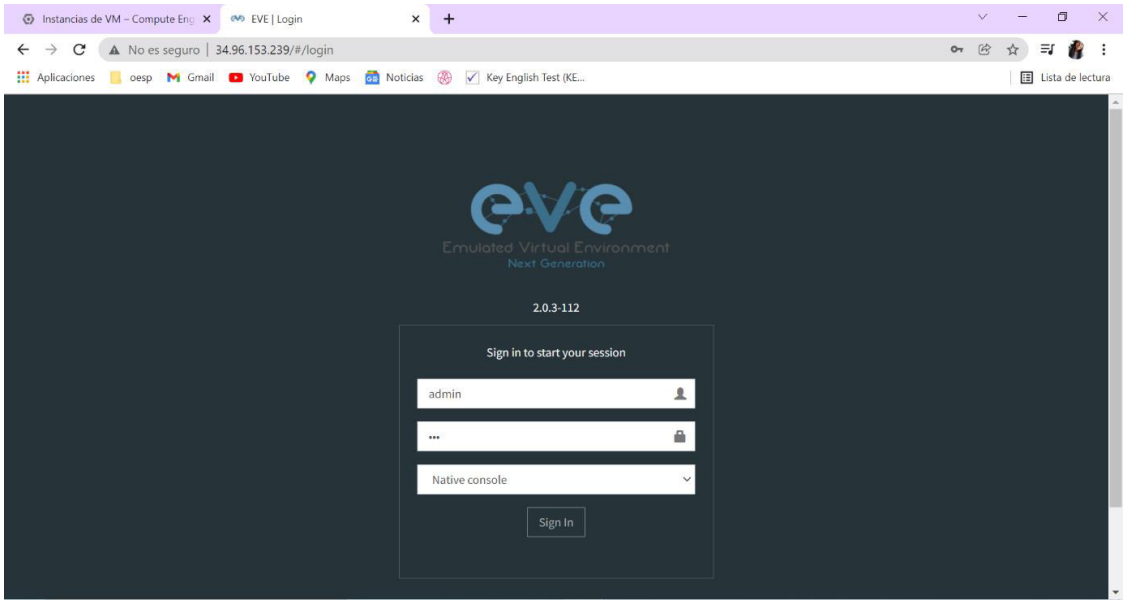

Figura 3.9 Pantalla de inicio de EVE-NG

3.5 Implementar las VPNs sobre MPLS en una red sobre c*loud computing*

Después de la investigación de VPNs sobre MPLS, se obtuvo que existen tanto VPNs de capa 2 como de capa 3 que están sobre MPLS, a partir de esto se realizaron cuatro redes, dos redes por cada capa. La primera red que se implementó ocupa VPNs de capa. La topología para implementar se encuentra en la Figura 3.10**.**

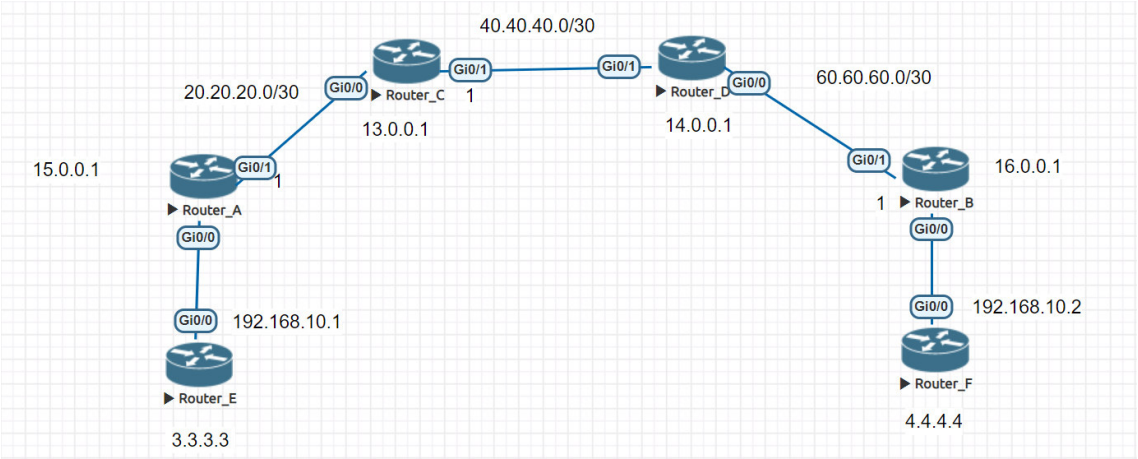

Figura 3.10 Topología de la red con VPN de capa 2

Para la configuración de las VPNs de capa 2, primero se levantó las interfaces tanto las conectadas como las de *loopback* en la Figura 3.11 se tiene de uno de los *routers* que se encuentran en el centro de la topología para su configuración.

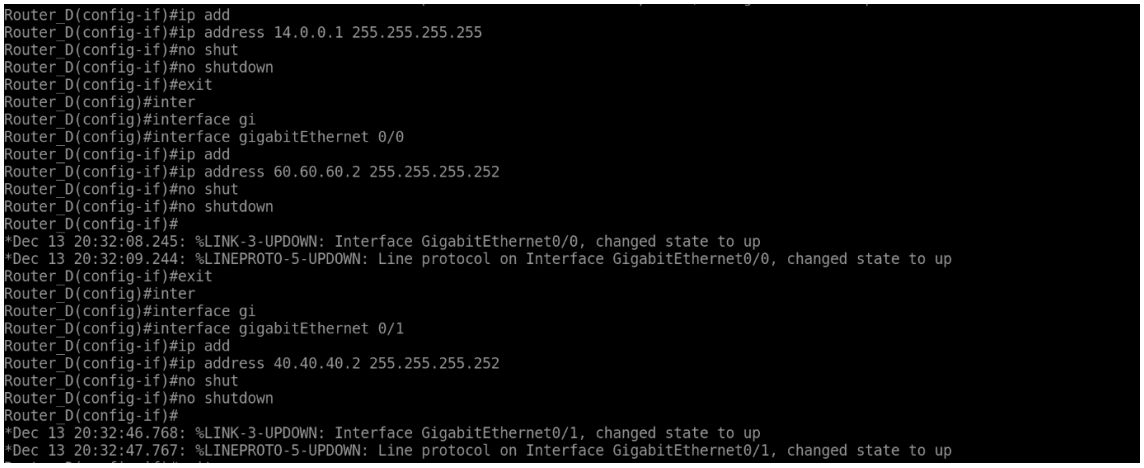

Figura 3.11 Configuración de las interfaces

A las interfaces directamente conectadas como de *loopback* de los dos *routers* centrales. Se ocupó el protocolo de enrutamiento EIGRP para declarar cada una de ellas, ya que los diferentes protocolos de enrutamiento permiten el funcionamiento de MPLS. Esta configuración se encuentra en la Figura 3.12**.**

Figura 3.12 Configuración del protocolo EIGRP

Al terminar con la configuración de los *routers* centrales, se debe comenzar con la configuración de los *routers* que tendrán las VPNs de capa 2, que en este caso son el *router* F y E, después de levantar sus interfaces conectadas y de *loopback*. Se declara sus direcciones en el protocolo de enrutamiento RIP versión 2, como muestra la Figura 3.13**.**

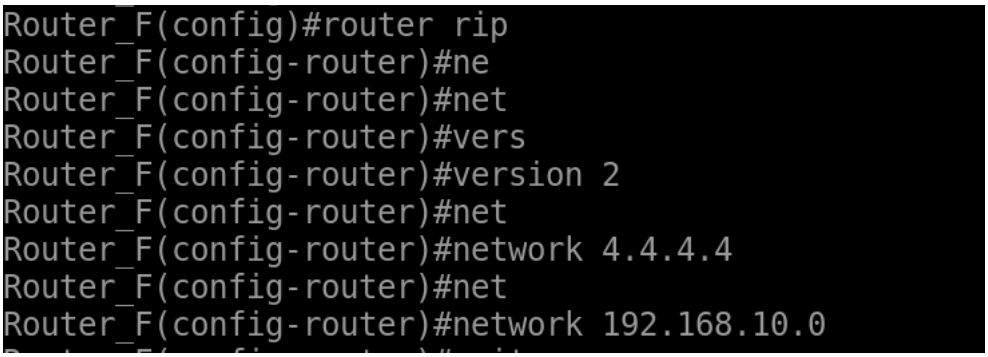

Figura 3.13 Configuración del protocolo RIPv2

Para la configuración de MPLS es necesario colocar que se ocupará LDP al igual que colocar el *router ID*, que viene cualquiera de las interfaces de *loopback* que se levantó y por último en cada interfaz que está directamente conectada se debe colocar que se ocupará MPLS, este procedimiento se encuentra en la Figura 3.14**.**

```
Router_D(config)#mpls label protocol ldp
Router_D(config)#mpls ld
Router_D(config)#mpls ldp ro
Router_D(config)#mpls ldp router-id loo
Router_D(config)#mpls ldp router-id loopback 0
Router_D(config)#inter
Router_D(config)#interface gi
Router_D(config)#interface gigabitEthernet 0/0
Router_D(config-if)#mp
Router_D(config-if)#mpls i
Router_D(config-if)#mpls ip
Router_D(config-if)#exit
Router_D(config)#inter
Router_D(config)#interface gi
Router_D(config)#interface gigabitEthernet 0/1
Router_D(config-if)#mp
Router_D(config-if)#mpls ip
Router_D(config-if)#mpls ip
Router_D(config-if)#
*Dec 13 20:35:44.144: %LDP-5-NBRCHG: LDP Neighbor 13.0.0.1:0 (1) is UP
```

Figura 3.14 Configuración de MPLS

Al terminar con la configuración de MPLS se comienza con la configuración de las VPNs de capa 2, todo este procedimiento se realiza en los *routers* A y B. Es necesario colocar el nombre que va a tener la VPN y dentro de esta configuración es donde se coloca el encapsulamiento de MPLS, que permite tener el camino para establecer conexión entre las distintas VPNs. En la Figura 3.15 se tendrá los comandos a ocupar para establecer el encapsulamiento.

```
Router_A(config)#pseudowire-class LSLINK_RouterE
Router_A(config-pw-class)#enca
Router_A(config-pw-class)#encapsulation mpls
```

Figura 3.15 Comando para encapsular MPLS

En los *routers* A y B se debe configurar la interfaz que se encuentra conectada con los *routers* de los extremos, donde se colocará la dirección IP del *router* del otro extremo con el nombre del *router* directamente conectado. Los comandos a utilizar se encuentran en la Figura 3.16.

```
Router_A(config)#interface gigabitEthernet 0/1
Router_A(config-if)#exit
Router_A(config)#interface gigabitEthernet 0/0
Router_A(config-if)# no xconnect 16.0.0.1 1 pw-class LSLINK_RouterE
Router_A(config-if)#xco
Router_A(config-if)#xconnect 16.0.0.1 ?
  <1-4294967295>  Enter VC ID value

Router_A(config-if)#xconnect 16.0.0.1 1 pw
Router_A(config-if)#xconnect 16.0.0.1 1 pw-class LSLINK_RouterE
```

Figura 3.16 Comandos para conectar las VPNs de capa 2

VPN3

La segunda red que se desarrolló ocupó VPN de capa3, para esta red se implementó la topología de la Figura 3.17**,** para esta red fue necesario añadir varias interfaces de loopback que iban a ser utilizadas tanto para la configuración de MPLS como de las VPNs, al igual se colocó un rango de etiquetas para los paquetes de MPLS, para en el momento de observar la tabla de enrutamiento se tenga un mayor orden de los paquetes y de donde proviene.

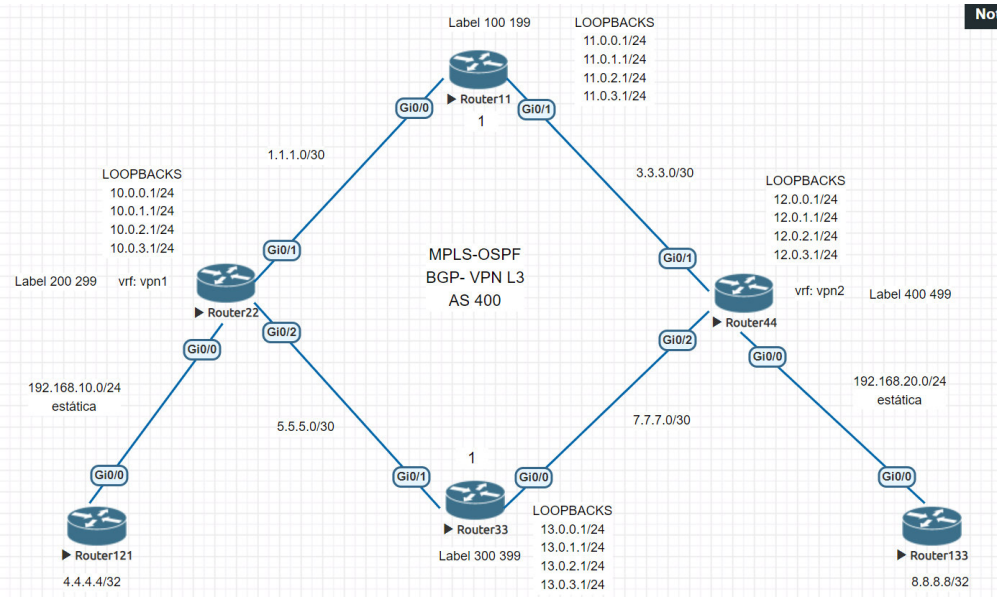

Figura 3.17 Topología de la red con VPN de capa 3

Para la configuración de MPLS se utilizó los mismos comandos que para la red de VPN de capa 2 que se encuentra en la Figura 3.18**,** pero ocupando el protocolo de enrutamiento OSPF**.** Al terminar con la configuración de MPLS, se comienza con el procedimiento a ocupar para las VPNs de capa 3 que al igual que las de capa 2 se conectarán de extremo a extremo, con la diferencia que se ocupará el protocolo de enrutamiento BGP. Para los *routers* 22 y 44 lo primero que se realizó fue colocar el

nombre a la VPN, al igual que el sistema autónomo del *router* que para esta red fue el 400 y por último fue necesario indicar que se exportarán e importarán datos a partir del protocolo BGP.

```
Router44(config)#router ospf 1
Router44(config-router)#net
Router44(config-router)#network 3.3.3.0 0.0.0.3 are
Router44(config-router)#network 3.3.3.0 0.0.0.3 area 0
Router44(config-router)#net
Router44(config-router)#network
*Dec 15 15:25:46.190: %OSPF-5-ADJCHG: Process 1, Nbr 11.0.3.1 on GigabitEthernet0/1 from LOADING to FULL, Loading Done
Router44(config-router)#network 7.7.7.0 0.0.0.3 are
Router44(config-router)#network 7.7.7.0 0.0.0.3 area 0
Router44(config-router)#net
Router44(config-router)#network 12.0.0.1 0.0.0.255 are
Router44(config-router)#network 12.0.0.1 0.0.0.255 area 0
Router44(config-router)#net
Router44(config-router)#network 12.0.1.1 0.0.0.255 area 0
Router44(config-router)#network 12.0.2.1 0.0.0.255 area 0
Router44(config-router)#network 12.0.3.1 0.0.0.255 area 0
```

Figura 3.18 Configuración del protocolo OSPF

La configuración que siguió se desarrolló igual en los *routers* 22 y 44, donde se ingresó a las interfaces conectadas con los *routers* del extremo. Para colocar el nombre de la VPN y su dirección IP correspondiente. Esta configuración se encuentra en la Figura 3.19**.**

```
Router22(config)#interface gigabitEthernet 0/0
Router22(config-if)#ip vr
Router22(config-if)#ip vrf for
Router22(config-if)#ip vrf forwarding vpn1
Router22(config-if)#ip add
Router22(config-if)#ip address 192.168.10.1 255.255.255.0
Router22(config-if)#no shut
Router22(config-if)#no shutdown
Router22(config-if)#
*Dec 15 15:35:51.946: %LINK-3-UPDOWN: Interface GigabitEthernet0/0, changed state to up
*Dec 15 15:35:52.947: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to up
```

Figura 3.19 Configuración de la VRF en las interfaces designadas

En los *routers* 22 y 44 se ingresó en la sesión de BGP, donde fue necesario quitar el *default* de IPV4 *Unicast,* ya que existe otro tipo de direccionamiento a través de las VPN de capa 3. Se estableció el vecino que se encuentra al otro extremo respectivamente tanto para el *route*r 22 como 44, que se encuentra en el mismo sistema autónomo. Ocupando el mismo vecino se actualizará las sesiones a través de la interfaz de *loopback*, esto permite tener un mayor control de las sesiones que se establezcan entre los extremos. Otra de las configuraciones necesarias es establecer que las VPNs de capa 3 al ocupar MPLS deben ser las VPNv4 *Unicast.*

Para terminar con la configuración de BGP es necesario activar al vecino que ya se estableció, al igual que indicar que el próximo salto que va a realizar va a ser al propio *router*, ya que si no se ocupa este comando el próximo salto va a ser inalcanzable. Cuando la configuración esté terminada en ambos *routers* debe salir un mensaje con

que el vecino se encuentra activo. Toda la configuración realizada se encuentra en la Figura 3.20**.**

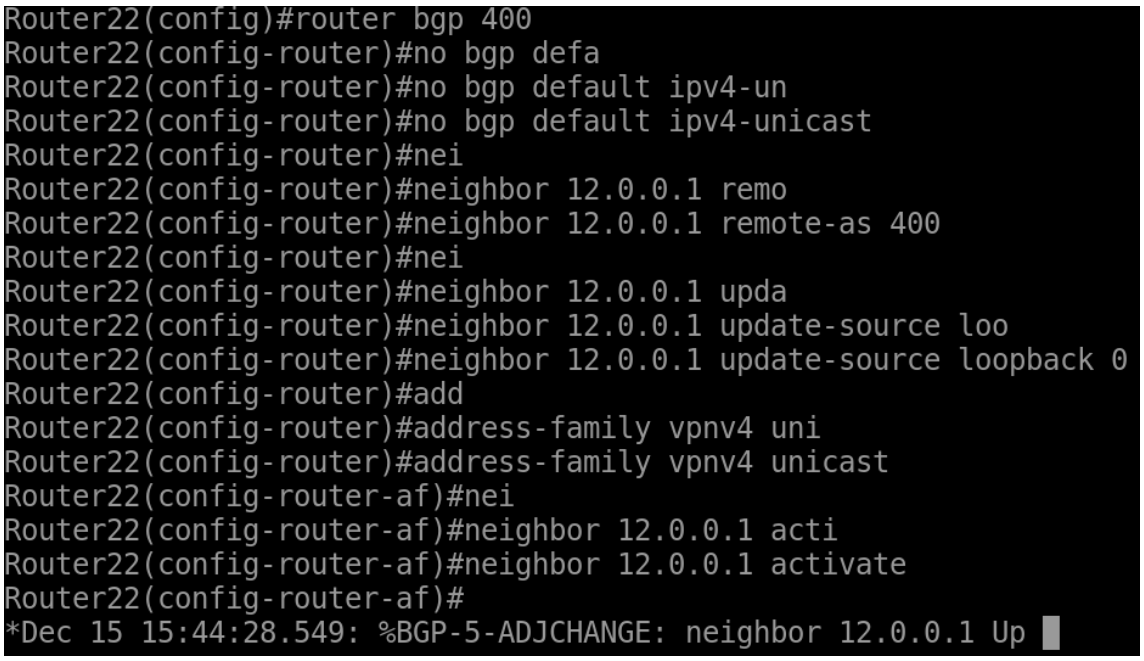

Figura 3.20 Configuración del protocolo BGP

En los *routers* de los extremos que son el 121 y 133, se levantó una interfaz de *loopback* con su respectiva dirección IP. Se realizó lo mismo, pero con la interfaz directamente conectada con el *router* 22 y 44 respectivamente. Para el caso de las VPNs de capa 3 es necesario ocupar a parte del enrutamiento dinámico, se ocupa el estático creando una ruta por defecto con el siguiente salto que viene a ser la dirección que ocupa el *router* 22 y 44. Estos comandos se encuentran en la Figura 3.21**.**

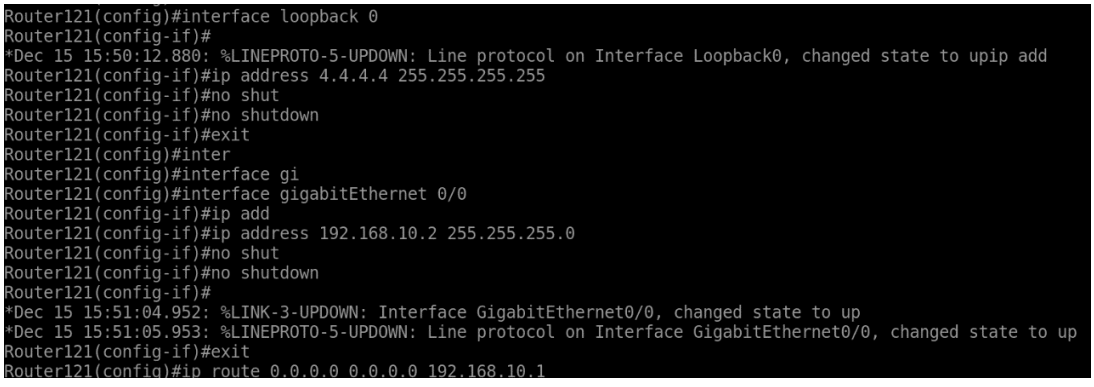

Figura 3.21 Configuración de interfaces de *loopback* y enrutamiento estático

Al concluir con la configuración de los *routers* de los extremos. Se vuelve a ingresar a los *routers* 22 y 44 para crear una ruta por defecto que ocupará la VRF, que necesitará

tanto la dirección de la red como el siguiente salto, todo esto se encuentra en la Figura 3.22.

```
Router22(config)#ip route vr
Router22(config)#ip route vrf vpn1 4.4.4.4 255.255.255.255 192.168.10.2
```

Figura 3.22 Ruta estática que utilizará la VRF

Es necesario redistribuir la ruta estática del VRF en la sesión BGP, para que esta pueda conocer y llegar de un extremo a otro ocupando las VPNs, los comando a ocupar están en la Figura 3.23.

```
Router22(config)#router bgp 400
Router22(config-router)#add
Router22(config-router)#address-family ipv4 vrf vpn1
Router22(config-router-af)#redis
Router22(config-router-af)#redistribute sta
Router22(config-router-af)#redistribute static
```

Figura 3.23 Redistribución de la ruta estática de la VRF

Se realizaron dos redes más, cada una de ellas con un tipo de VPNs sobre MPLS. Como los pasos a realizar llegan a ser los mismos, se colocó la topología de cada una de ellas en la Figura 3.24 que es la red de VPN capa 3, mientras que la Figura 3.25 es la red de VPN de capa 2.

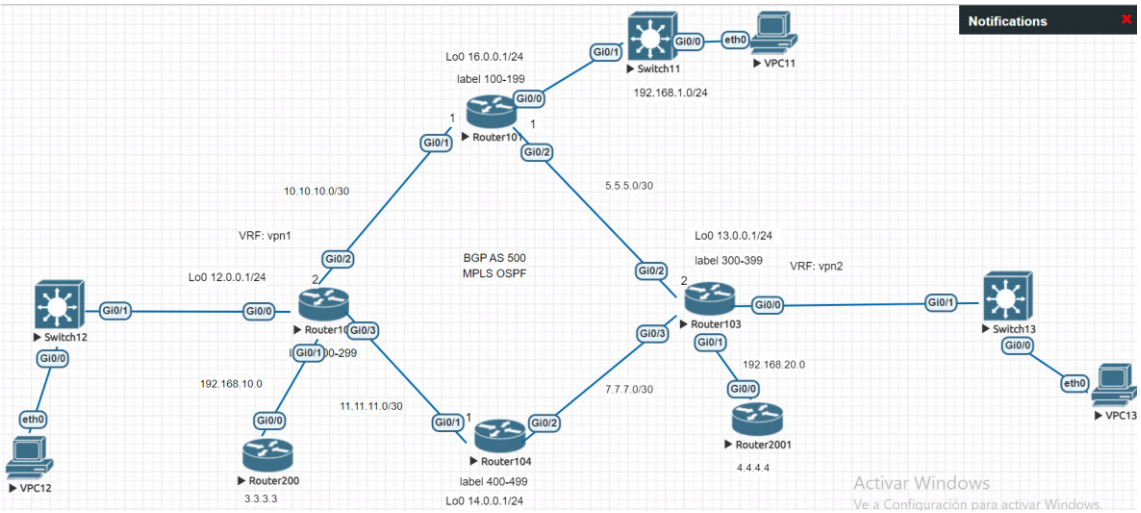

Figura 3.24 Topología de la segunda red con VPN de capa 3

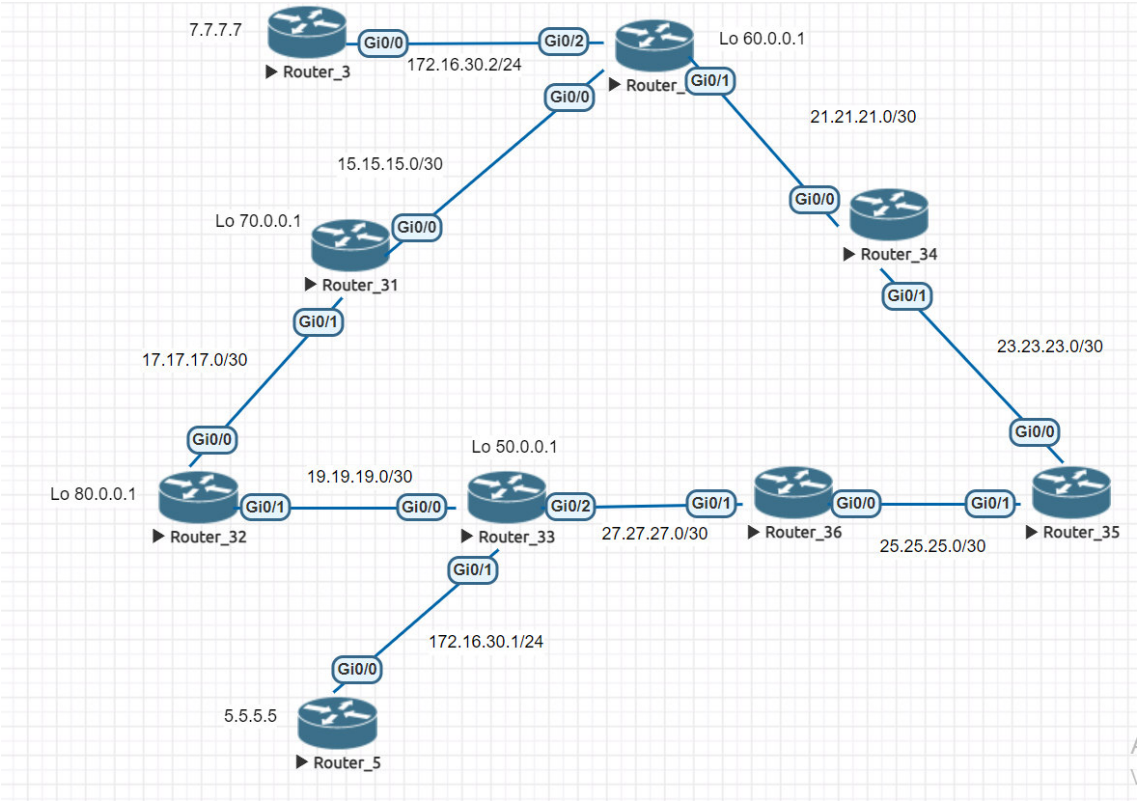

Figura 3.25 Topología de la segunda red con VPN de capa 2

3.6 Realizar pruebas de funcionamiento

Para las pruebas de funcionamiento, se utilizó comandos como *ping*, *traceroute* al igual que la tabla de MPLS como de enrutamiento. Para el caso de las redes que se implementaron con VPN de capa 2, se obtuvo la tabla de MPLS con capa 2 que está en la Figura 3.26**.**

```
Router_B#show mpls l2 vc 1

Local intf     Local circuit              Dest address    VC ID      Status
-------------  -------------------------- --------------- ---------- ----------
Gi0/0          Ethernet                   15.0.0.1        1          UP
```

Figura 3.26 Tabla de la VPN sobre MPLS en capa 2

Se obtuvo la tabla de enrutamiento de la Figura 3.10 con los protocolos ocupados en estas redes que son RIPv2 como EIGRP, esta tabla se encuentra en la Figura 3.27**.**

```
Router_E#show ip route
Codes: L - local, 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, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override, p - overrides from PfR

Gateway of last resort is not set

      3.0.0.0/32 is subnetted, 1 subnets
C        3.3.3.3 is directly connected, Loopback0
R     4.0.0.0/8 [120/1] via 192.168.10.2, 00:00:20, GigabitEthernet0/0
      192.168.10.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.10.0/24 is directly connected, GigabitEthernet0/0
L        192.168.10.1/32 is directly connected, GigabitEthernet0/0
```

Figura 3.27 Tabla de enrutamiento de los protocolos ocupados en la red

Por último, se obtuvo la tabla de solo MPLS con el túnel, su etiqueta y su próximo salto, esta tabla está en la Figura 3.28**.**

```
Router_A#show mpls forwarding-table
Local      Outgoing   Prefix           Bytes Label   Outgoing   Next Hop
Label      Label      or Tunnel Id     Switched      interface
100        Pop Label  13.0.0.1/32      0             Gi0/1      20.20.20.2
101        17         14.0.0.1/32      0             Gi0/1      20.20.20.2
102        Pop Label  40.40.40.0/30    0             Gi0/1      20.20.20.2
103        16         60.60.60.0/30    0             Gi0/1      20.20.20.2
104        19         16.0.0.1/32      0             Gi0/1      20.20.20.2
105        No Label   l2ckt(1)         27715         Gi0/0      point2point
```

Figura 3.28 Tabla de MPLS ocupando VPN de capa 2

Se realizó pruebas de conexión con el comando *ping*, como ya se había indicado el resultado de esta prueba fue satisfactorio, ya que todos los paquetes de extremo a extremo fueron recibidos como se puede observar en la Figura 3.29**.**

```
Router_E#ping 4.4.4.4
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 4.4.4.4, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 12/15/20 ms
```

Figura 3.29 Comprobación con el comando *ping*

Para poder observar la VRF y la dirección que ocupa para llegar a su destino se utilizó el comando *traceroute*, como se muestra en la Figura 3.30**.**

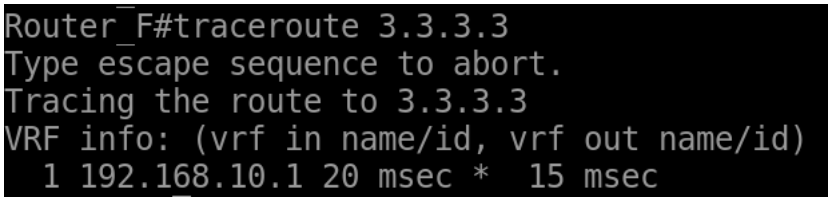

Figura 3.30 Comprobación con el comando t*raceroute*

Para el caso de las redes con VPN de capa 3 se realizó pruebas con el comando ping de un extremo a otro, para comprobar que existe conexión entre ellas. Se puede observar en la Figura 3.31**.**

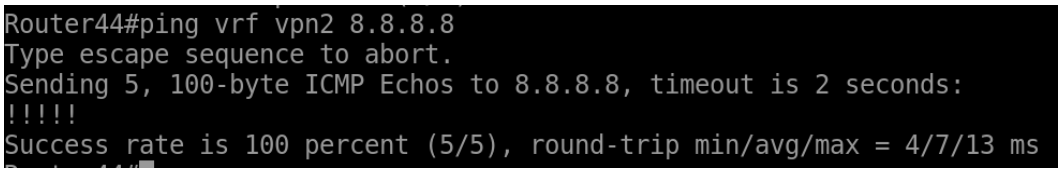

Figura 3.31 Comprobación con el comando p*ing* la red de VPN de capa 3

Existe conexión desde el *router* del extremo al otro, mientras que en la Figura 3.32 se realizó la misma prueba, pero entre los *routers* directamente conectados a través de la VPN.

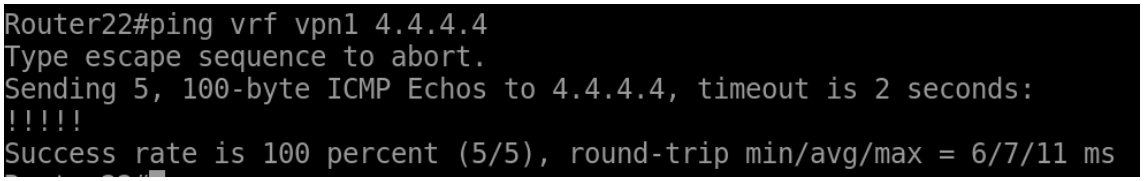

Figura 3.32 Prueba en la misma

La tabla de enrutamiento de la topología de la red de VPN de capa 3, que se encuentra en la Figura 3.24. Dentro de esta tabla se visualiza las direcciones que conoció por los diferentes protocolos que se llega a ocupar tanto para MPLS como VPNs, todo esto se encuentra en la Figura 3.33**.**

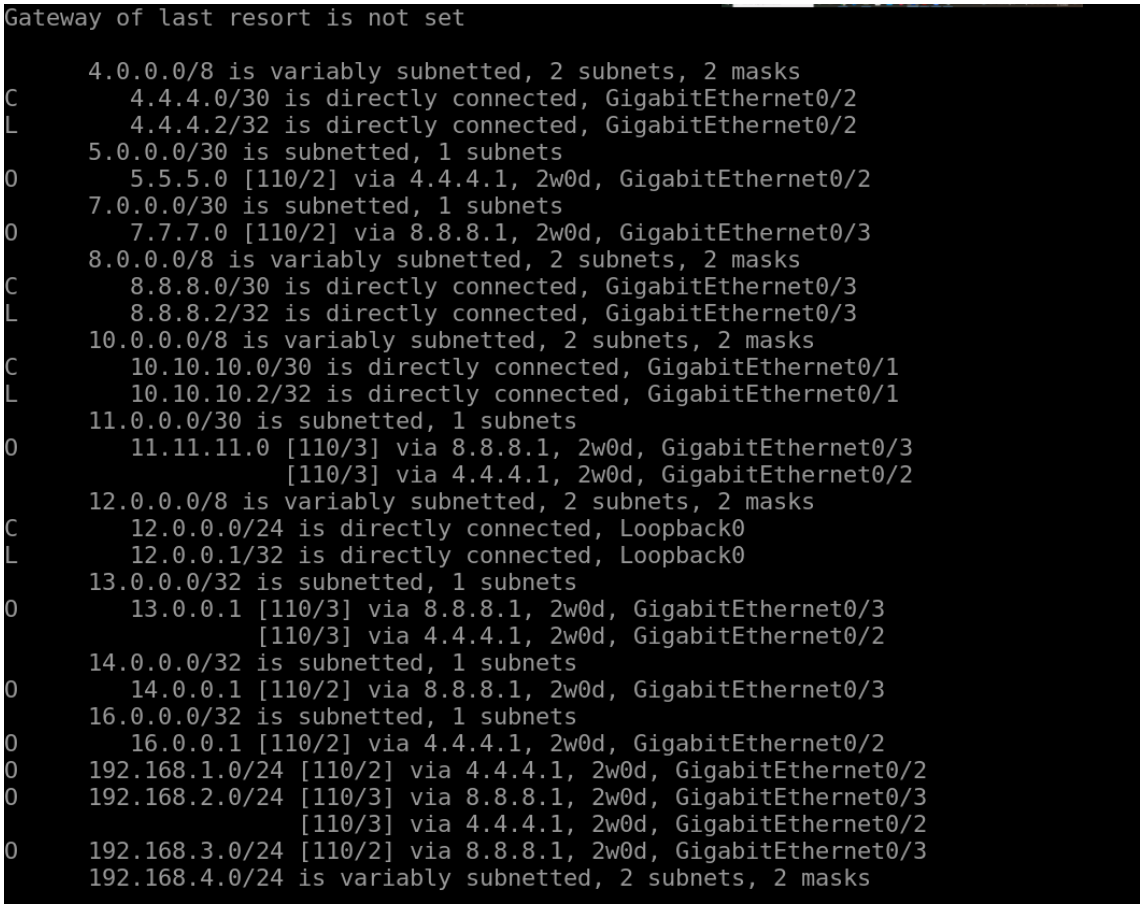

Figura 3.33 Tabla de enrutamiento de la red con VPN de capa 3

En la Figura 3.34 se tiene la tabla de MPLS de la red de la Figura 3.24**.** Que cuenta con las etiquetas designadas que se colocó en la configuración. Al igual que la dirección IP, por la cual viaja el paquete.

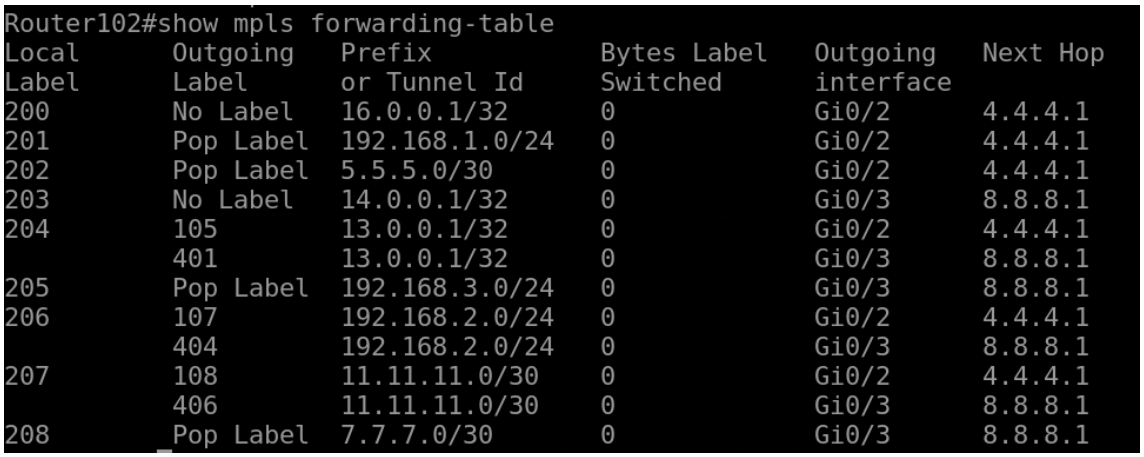

Figura 3.34 Tabla de MPLS de la red con VPN de capa 3

En la Figura 3.35 se tiene la tabla de VPNs de la red de la Figura 3.17**.** Donde se conoce qué protocolo está ocupando el nombre de la VRF y la interfaz que ocupó para transferir los datos.

```
Router22#show vrf
  Name                             Default RD          Protocols   Interfaces
  vpn1                             400:1               ipv4        Gi0/0
```

Figura 3.35 Tabla de las VRF en la red

En la Figura 3.36 se tiene la implementación de *ping* de la red de la Figura 3.24 donde fue necesario colocar el nombre de la VRF y la dirección de esta, para establecer la conexión, mientras que en la Figura 3.37 se utilizó el comando *traceroute* que permitió observar por donde viaja el paquete de la VPN sobre MPLS. Estas dos comprobaciones fueron realizadas de un extremo a otro.

```
Router103#ping vrf vpn2 4.4.4.4
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 4.4.4.4, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 14/20/40 ms
```

Figura 3.36 *Ping* realizado en la red de VPN de capa 3

```
Router103#traceroute vrf vpn2 4.4.4.4
Type escape sequence to abort.
Tracing the route to 4.4.4.4
VRF info: (vrf in name/id, vrf out name/id)
  1 192.168.20.2 15 msec *  13 msec
```

Figura 3.37 *Traceroute* realizado en la red de VPN de capa 3

4 CONCLUSIONES

- A partir de la implementación del trabajo de titulación, se puede analizar que la utilización de las VPNs es una manera segura de establecer una conexión remota de extremo a extremo, ya que cuenta con el cifrado de información al momento que viaje por el Internet.
- A partir de la investigación se pudo observar que las VPNs de capa 2 son fáciles de implementar, ya que no requieren de protocolos de enrutamiento en específico o demasiada configuración para la creación de la VPN. En cambio, las VPNs de capa 3 a parte de ocupar BGP, se debe realizar configuraciones extra dentro de este protocolo, al igual que ocupar enrutamiento estático.
- Dentro del ámbito laboral las VPNs pueden ser implementadas para la conexión remota de los trabajadores, ya que se tiene que en época de pandemia muchas empresas tuvieron que implementar el teletrabajo y la utilización de las VPNs les permite tener una conexión segura y cifrado y evitar inconvenientes como el robo de información al ocupar la conexión remota.
- En la configuración de MPLS, se pudo observar que lo primero a realizar es habilitar el protocolo LDP que permite distribuir las etiquetas, que también existen comandos que permiten agregar el rango de las etiquetas en cada *router,* lo cual admite tener un mayor control de las etiquetas y conocer que *router* está transmitiendo los datos.
- En los *routers* que solo manejan VPNs de capa 2 y 3 se puede observar que sus tablas de enrutamiento solo cuentan con las interfaces directamente conectadas y la interfaz de la VPN del extremo que conoce por el protocolo de enrutamiento ocupado para su configuración.
- Al momento de ocupar el comando *traceroute* para conocer el camino por donde el paquete viaja se puede observar que solo tiene una ruta que es la que conoció por VPNs sobre MPLS, esto permite que el paquete llegue más rápido debido a que no viaja por toda la red para llegar a su destino si no que ya conoce la ruta establecida por MPLS.
- Al ocupar el comando *ping* para comprobar el funcionamiento de la red, se puede observar que en el caso de las VPNs de capa 2 no es necesario colocar el nombre de la VPN, solo con tener la dirección se establece conexión, mientras que en el caso de las VPNs de capa 3 es necesario conocer el nombre de la VRF y su dirección para establecer conexión sí este dato extra nunca se podrá

conectar estas dos VPNS, al necesitar el nombre de la VRF estas VPNs aumentan la seguridad de cualquier usuario se conecte a su VPN.

- En las tablas de enrutamiento de los *routers* que no cuentan con la configuración de las VPNS, se puede observar que no tienen entre sus direcciones conocidas por los protocolos dinámicos a las direcciones IP de las VPNs, ya que estos *routers* no pueden conectarse a las VPNs al menos que se realice la configuración dependiendo del tipo de VPN que se necesite.

5 RECOMENDACIONES

- Para la realización de las redes, fue necesario realizar una investigación de las imágenes de los equipos a ocupar. Debido a que dependiendo de la imagen ISO, esta puede o no tener permitido la configuración de MPLS o de las VPNS. Debido a esto la información de las ISO, es muy necesario para evitar ocupar una imagen que no permita estos protocolos y volver a tener que buscar otra imagen que sí permita realizar esta implementación.
- Al momento de realizar la implementación de redes tanto con VPNs de capa 2 y 3 es necesario la utilización de interfaces de *loopback,* ya que durante la configuración de MPLS y VPNs estas direcciones sirven para establecer los vecinos de extremo a extremo o para que la red conozca que esta dirección es el *router-ID* de MPLS.
- Al momento de implementar las redes de VPN de capa 3, no realizar la configuración de MPLS y de las VPNs con el mismo protocolo de enrutamiento dinámico, ya que se llega a cometer errores por lo complejo de su configuración. Es recomendable configurar MPLS con un protocolo, mientras que las VPNs con el protocolo BGP, ya que después se realiza una redistribución de la dirección que ocupa la VPN, para que ambas VPRN se conozcan entre sí.
- En el caso de existir fallas con las interfaces de *loopback* en la configuración de BGP en las redes de VPNs de capa3. Cambiar la máscara de la dirección IP por una máscara 32.
- Si llegan a existir inconvenientes al instalar EVE-NG sobre Google Cloud, se puede tener el problema en la región seleccionada, se puede cambiar la zona geográfica, pero siempre verificando el número de máquinas que soporta cada región.
- Al momento de configurar las redes de VPNs sobre MPLS, siempre contar con varias interfaces de *loopback,* ya que puede existir inconvenientes al momento de realizar la configuración de MPLS o de las VPNs.
- Al indicar las direcciones IP de las VPNs de capa 2, estas deben encontrarse en la misma red, ya que si llega a estar en diferentes redes estas no se podrán conectar entre sí.
- En el caso de necesitar una dirección por defecto, esta se realiza con enrutamiento estático y es necesario colocar una dirección y una máscara de cero y la dirección del próximo salto.

6 REFERENCIAS BIBLIOGRÁFICAS

[1] I. Rodriguez, «Scrib,» Enero 2002. [En línea]. Available: https://es.scribd.com/document/173398855/SIMULACION-EN-REDES-docx. [Último acceso: 24 Noviembre 2021].

[2] V. R. C. Panduro, «Simulación y Emulación de la Red Universitaria Nacional de Chile y el nivel de comprensión del funcionamiento de redes avanzadas,» Escuela Profesional de Ingeniería en Informática y Sistemas, Chile , 2018.

[3] Acronis, «Acronis,» 2010. [En línea]. Available: https://www.acronis.com/en-us/articles/google-cloud-platform/. [Último acceso: 25 Noviembre 2021].

[4] Citelia, «Citelia,» 8 Abril 2021. [En línea]. Available: https://citelia.es/blog/que-es-una-red-mpls-y-como-funciona/. [Último acceso: 25 Noviembre 2021].

[5] Entel, «CE Entel,» 2021. [En línea]. Available: https://ce.entel.cl/pymes/articulos/red-mpls-que-es-beneficios/. [Último acceso: 25 Noviembre 2021].

[6] Juniper, «juniper,» 4 Enero 2021. [En línea]. Available: https://www.juniper.net/documentation/mx/es/software/junos/mpls/topics/topic-map/ldp-overview.html. [Último acceso: 2 Enero 2022].

[7] G. López, «Pruebas de escala de VPNs capa 2 y 3 para la implementación legada basada en MPLS,» Universidad de la República, Montevideo.

[8] C. Cabrera, «Cesar Cabrera,» 6 Mayo 2018. [En línea]. Available: https://cesarcabrera.info/que-es-eve-ng-un-nuevo-emulador-para-redes/. [Último acceso: 28 Noviembre 2021].

[9] «EVE-NG,» 2021. [En línea]. Available: https://www.eve-ng.net/index.php/documentation/. [Último acceso: 28 Noviembre 2021].

[10] Teletrónika, «telectronika,» 29 Abril 2018. [En línea]. Available: https://www.telectronika.com/articulos/ti/que-es-gns3/. [Último acceso: 29 Noviembre 2021].

[11] pnetlab, «pnetlab,» 2021. [En línea]. Available: https://pnetlab.com/pages/main. [Último acceso: 05 Enero 2022].

[12] R. Bhardwaj, «ipwithease,» 2020. [En línea]. Available: https://ipwithease.com/gns3-vs-eve-ng-vs-virl/. [Último acceso: 2 Enero 2022].

[13] S. Schmidt, «aws amazon,» Marzo 2006. [En línea]. Available: https://aws.amazon.com/es/products/?aws-products-all.sort-by=item.additionalFields.productNameLowercase&aws-products-all.sort-order=asc&awsf.re%3AInvent=*all&awsf.Free%20Tier=*all&awsf.tech-category=*all. [Último acceso: 24 Noviembre 2021].

[14] Incentro, «incentro,» 19 Agosto 2020. [En línea]. Available: https://www.incentro.com/es-es/blog/stories/que-es-google-cloud-platform/. [Último acceso: 9 Enero 2022].

[15] S. Services, «S&M Services,» [En línea]. Available: https://sm-services.es/que-es-openshift/. [Último acceso: 28 Noviembre 2021].

[16] Tecon, «tecon,» 17 Enero 2020. [En línea]. Available: https://www.tecon.es/que-es-microsoft-azure-como-funciona/. [Último acceso: 27 Noviembre 2021].

[17] Cisco, «cisco,» [En línea]. Available: https://www.cisco.com/c/en/us/products/routers/index.html#~resources. [Último acceso: 06 Diciembre 2021].

[18] NordVPN, «nordvpn,» 15 Enero 2015. [En línea]. Available: https://nordvpn.com/es/what-is-a-vpn/. [Último acceso: 24 Noviembre 2021].

[19] D. VMware, «docs vmware,» 31 Mayo 2019. [En línea]. Available: https://docs.vmware.com/es/VMware-NSX-Data-Center-for-vSphere/6.4/com.vmware.nsx.troubleshooting.doc/GUID-BF355C21-93FF-41FC-97AC-EA45B10130A6.html. [Último acceso: 24 Noviembre 2021].

[20] I. P. Castellar, «el puig,» 23 Diciembre 2018. [En línea]. Available: https://elpuig.xeill.net/Members/vcarceler/articulos/qemu. [Último acceso: 31 Enero 2022].

[21] D. Rosales, «delfirosales,» 5 Mayo 2009. [En línea]. Available: https://delfirosales.blogspot.com/2009/05/instalacion-y-configuracion-de.html. [Último acceso: 31 Enero 2022].

7 ANEXOS

ANEXO I

CERTIFICADO DE ORIGINALIDAD

Quito, 21 de febrero de 2022

De mi consideración:

Yo, FERNANDO VINICIO BECERRA CAMACHO, en calidad de director del Trabajo de Integración Curricular titulado IMPLEMENTACIÓN DE LA RED CON VIRTUALIZACIÓN EN LA *CLOUD COMPUTING* asociado a la VIRTUALIZACIÓN DE UNA RED MPLS CON VPNS EN UNA *CLOUD COMPUTING* elaborado por la estudiante ANA PAULA PANCHES MORA de la carrera en TECNOLOGÍA SUPERIOR EN REDES Y TELECOMUNICACIONES, certifico que he empleado la herramienta Turnitin el informe para la revisión de originalidad del documento escrito completo producto del Trabajo de Integración Curricular indicado.

El documento escrito tiene un índice de similitud de 8%.

Es todo cuanto puedo certificar en honor a la verdad, pudiendo el interesado hacer uso del presente documento para los trámites de titulación.

NOTA: Se adjunta el informe generado por la herramienta Turnitin

https://epnecuador-my.sharepoint.com/:b:/g/personal/fernando_becerrac_epn_edu_ec/EZF0_Jy0AbNFuNGku93u4vUBYNZVoVMQy5h7LkQmnrUX1w?e=ggTRwk

Atentamente,

Fernando Vinicio Becerra Camacho

Profesor ocasional a tiempo completo

Escuela de Formación de Tecnólogos

ANEXO II

HTTPS://EPNECUADOR-MY.SHAREPOINT.COM/:V:/G/PERSONAL/ANAPAULA_PANCHES_EPN_EDU_EC/EQVNA4VKCUJCTMTWZAENC3MBNF3T4CPBK7XMPZOWNNFH0G