



**UNIVERSIDAD
DE ANTIOQUIA**



**SOFTWARE DE CONFIGURACIÓN PARA LOS SERVICIOS IP NEXT GENERATION Y
CARRIER ETHERNET EN LA RED MPLS HUAWEI DE INTERNEXA COLOMBIA**

Camilo Enrique Farelo Panesso

Informe de práctica como requisito para optar al título de:
Ingeniero de Telecomunicaciones

Asesores:

Erwin Alexander Leal Piedrahíta – Ingeniero Electrónico

Julián Esteban Montoya Gallego – Ingeniero de Telecomunicaciones

Camilo Pérez Correa – Ingeniero Electrónico

Universidad de Antioquia

Facultad de Ingeniería, Departamento de Ingeniería Electrónica y
Telecomunicaciones.

Medellín, Colombia

2021

Resumen

Algunos de los servicios prestados por Internexa Colombia son Carrier Ethernet (CE) e IP Next Generation (IPNG), los cuales son entregados por medio de su red MPLS (MultiProtocolo Label Switching) de Huawei. Desde hace más de 20 años estos servicios se han configurado de manera manual por medio de la interfaz de línea de comandos (CLI – Command Line Interface). Este esquema de configuración requiere que la persona encargada de dicha tarea tenga un amplio conocimiento de la topología de red, los protocolos L2-L3 que requiera el servicio y los comandos a utilizar dependiendo de la plataforma en la que se va a configurar el servicio. Este proceso es algo dispendioso, incluso la configuración de cada servicio toma alrededor de 30 a 45 minutos, sin contar los posibles imprevistos a la hora del aprovisionamiento de un servicio, sin mencionar la tasa de conexión de nuevos servicios tiene una tendencia al alza en la compañía.

El NOC (Network Operation Center) Latam hace poco menos de un año desarrolló una herramienta de configuración automática en Python, lo cual en su momento representaba una gran solución en cuanto a optimización de tiempo y recurso humano. Sin embargo, como consecuencia del cambio de políticas de configuración al interior de la organización, dicha aplicación pasó a ser obsoleta a pesar de su reciente desarrollo.

En este informe se documenta el proceso realizado para la modificación de la herramienta y la agregación de nuevas funcionalidades, lo cual permitió a los Ingenieros de Configuración la reducción de tiempos y de errores humanos para la entrega de los servicios CE e IPNG.

Tabla de Contenido

| | |
|--|----|
| 1. Introducción | 8 |
| 2. Objetivos | 10 |
| 3. Marco Teórico | 11 |
| 3.1 Modelo OSI..... | 11 |
| 3.1.1. Capa Física | 12 |
| 3.1.2. Capa de Enlace..... | 12 |
| 3.1.3. Capa de Red | 13 |
| 3.1.4. Capa de Transporte..... | 14 |
| 3.1.5. Capa de Sesión | 14 |
| 3.1.6. Capa de Presentación | 14 |
| 3.1.7. Capa de Aplicación..... | 15 |
| 3.2 Tecnología MPLS | 15 |
| 3.3 Red MPLS Huawei Colombia | 16 |
| 3.3.1. Switches Agile de la serie 12700 Huawei | 18 |
| 3.3.2. Switches de enrutamiento S6700 | 19 |
| 3.3.3. Switches de enrutamiento S9300 | 20 |
| 3.3.4. Servicio Carrier Ethernet | 21 |
| 3.3.5. IP Next Generation | 21 |
| 3.4 VLAN | 22 |
| 3.5 Subinterfaz..... | 24 |
| 3.6 Gestión de redes de Telecomunicaciones | 24 |
| 3.7 Protocolo SSH..... | 25 |
| 3.8 Descripción de Desarrollo de software..... | 26 |
| 3.8.1. Norma ISO/IEC/IEEE 42010..... | 26 |
| 3.8.2. Modelo de Kruchten 4+1 | 28 |
| 3.8.2.1. Vista Lógica | 28 |
| 3.8.2.2. Vista de Despliegue | 29 |
| 3.8.2.3. Vista de Procesos..... | 29 |
| 3.8.2.4. Vista Física..... | 29 |
| 3.8.2.5. "+1" Vista de Escenarios | 29 |
| 4. Metodología..... | 29 |
| 4.1 Estudio de conceptos básicos | 30 |
| 4.2 Contextualización en el aprovisionamiento de servicios CE e IPNG. | 31 |

| | |
|---|----|
| 4.2.1. Datos Generales..... | 32 |
| 4.2.2. Datos del equipo..... | 32 |
| 4.3 Construcción de la nueva versión del software..... | 33 |
| 4.3.1. Estudio del funcionamiento de la primera versión del software..... | 33 |
| 4.3.1.1. Carga de información desde la base de datos..... | 33 |
| 4.3.1.2. Interfaz gráfica de Usuario (GUI) | 35 |
| 4.3.1.3. Conexiones seguras | 38 |
| 4.3.2. Aplicación de los cambios en el software | 40 |
| 4.3.2.1. GUI | 41 |
| 4.3.2.2. Implementación de Servicios | 44 |
| 4.3.2.3. Escritura de Código..... | 45 |
| 4.3.3. Descripción del Software Desarrollado..... | 46 |
| 4.3.3.1. Vista Lógica | 47 |
| 4.3.3.2. Vista de Despliegue | 48 |
| 4.3.3.3. Vista de Procesos..... | 49 |
| 4.3.3.4. Vista Física..... | 50 |
| 4.3.3.5. +1 Vista de Escenarios..... | 51 |
| 4.4 Aprobación y entrega del software | 52 |
| 4.4.1. Pruebas de usuario..... | 52 |
| 4.4.2. Realización de los manuales | 52 |
| 4.4.2.1. Modificación del manual de configuración | 52 |
| 4.4.2.2. Modificación del manual de usuario | 53 |
| 4.4.3. Toma de tiempos | 53 |
| 4.4.4. Entrega oficial del software | 53 |
| 5. Resultados y análisis..... | 54 |
| 5.1 Tiempos de Configuración | 54 |
| 5.1.1. Tiempo de establecimiento de un servicio CE VLL | 54 |
| 5.1.2. Tiempos de establecimiento para un servicio IPNG | 55 |
| 5.2 Experiencia de Usuario..... | 57 |
| 5.3 Escalabilidad | 57 |
| 5.4 Mejoras en los servicios entregados..... | 57 |
| 6. Conclusiones | 58 |
| 7. Referencias Bibliográficas | 59 |

Índice de Figuras

| | |
|--|----|
| Figura 1: Estructura general del modelo OSI. | 11 |
| Figura 2: Elementos correspondientes a la capa física del modelo OSI. | 12 |
| Figura 3: Campos contenidos en una trama Ethernet. | 13 |
| Figura 4: Funciones realizadas dentro de la capa de red. | 13 |
| Figura 5: Funciones de la capa de sesión dentro del modelo OSI. | 14 |
| Figura 6: Funciones de la capa de presentación dentro del modelo OSI. | 15 |
| Figura 7: Protocolos de la capa 7 del Modelo OSI. | 15 |
| Figura 8: Estructura de una red MPLS. | 16 |
| Figura 9: Cabecera MPLS dentro de un paquete. | 16 |
| Figura 10: Topología de red en anillo de Internexa Colombia. | 18 |
| Figura 11: Switches de la línea 12700. | 19 |
| Figura 12: Modelos de la serie S6700. | 20 |
| Figura 13: Modelos de la serie S9300. | 20 |
| Figura 14: Ejemplo de servicio CE en Internexa. | 21 |
| Figura 15: Ejemplo de un servicio IPNG para un cliente en Internexa. | 22 |
| Figura 16: Cables submarinos de Internexa Colombia. | 22 |
| Figura 17: Etiqueta VLAN dentro de una trama Ethernet. | 23 |
| Figura 18: Ejemplo de clientes conectados por VLAN. | 24 |
| Figura 19: Proceso de cifrado en una conexión SSH. | 25 |
| Figura 20: Esquema de autenticación remota usada en Internexa. | 26 |
| Figura 21: Representación Simple del estándar ISO/IEC/IEEE 42010. | 27 |
| Figura 22: Vistas de Kruchten. | 28 |
| Figura 23: Datos esenciales para configurar cualquier servicio. | 31 |
| Figura 24: Tabla que contiene información de los equipos Huawei. | 34 |
| Figura 25: Tabla que contiene información de los equipos Alcatel. | 34 |
| Figura 26: Equipos protegidos Alcatel. | 35 |
| Figura 27: Equipos protegidos Huawei. | 35 |
| Figura 28: Historial de configuraciones. | 35 |
| Figura 29: Pestañas de la primera versión del software. | 35 |
| Figura 30: Boceto de la pestaña Principal de la primera versión del software. | 36 |
| Figura 31: Apariencia real del boceto sugerido por el NOC. | 36 |
| Figura 32: Datos necesarios para un servicio IPNG. | 37 |
| Figura 33: Esquema de funcionamiento shell-based. | 39 |
| Figura 34: GUI principal con modificaciones. | 42 |
| Figura 35: Cambios en el menú de IPNG. | 43 |
| Figura 36: Cambios en la GUI en VSI. | 43 |
| Figura 37: Ventana de configuración modificada. | 44 |
| Figura 38: Ventana de Historial modificada. | 44 |
| Figura 39: Verificación de la convención PEP8 del código. | 46 |
| Figura 40: Ejemplo de la convención PEP257. | 46 |
| Figura 41: Vista Lógica del software. | 48 |
| Figura 42: Vista de Despliegue del Software. | 48 |
| Figura 43: Diagrama de actividades del software. | 50 |

| | |
|--|----|
| Figura 44: Vista Física de la arquitectura..... | 51 |
| Figura 45: Vista de Escenarios del Software..... | 51 |

Índice de Tablas

| | |
|--|----|
| Tabla 1: Listado de modificaciones realizadas por el software..... | 40 |
| Tabla 2: Resultados de los servicios CE VLL. | 54 |
| Tabla 3: Tiempos de Configuración IPNG. | 56 |
| Tabla 4: Porcentajes obtenidos con respecto al tiempo manual..... | 56 |

1. Introducción

Internexa es una empresa filial del grupo empresarial colombiano ISA, que cuenta con 20 años de experiencia en el mercado de las telecomunicaciones entregando soluciones tecnológicas seguras, confiables e innovadoras dirigidas a operadores de telecomunicaciones, empresas privadas e instituciones gubernamentales. La compañía conecta a varios países de Latinoamérica (Colombia, Argentina, Brasil, Perú, Chile y Centroamérica) con Estados Unidos por medio de un sistema de redes en las que se incluyen redes metropolitanas, nacionales e internacionales. Esta de red cuenta con tecnología de punta y posee más de 54.000 kilómetros de fibra óptica que permiten conectar a más de 868 clientes a través de 257 puntos de conexión.

Para prestar servicios en Colombia, Internexa cuenta con la topología de red MPLS (MultiLabel Switching Protocol) Huawei, la cual provee 6 anillos interconectados y alrededor de 37 elementos de red de backbone ubicados en sectores estratégicos a nivel nacional. Al interior de esta red se brindan servicios como Carrier Ethernet (CE), que es un servicio de capa 2 prestado a las empresas para interconectar sus sedes de manera local; e IP Next Generation (IPNG), que consiste en un servicio capa 3 que provee un canal para conectarse a internet y acceder a los CDN (Content Distribution Network) locales con los cuales la compañía cuenta actualmente.

El área encargada dentro de la empresa de realizar estas configuraciones corresponde al NOC (Network Operation Center) Latam, que cuenta con profesionales con énfasis en Ingeniería de tráfico, aprovisionamiento de servicios, de monitoreo y supervisión de la red, entre otros. Los ingenieros de configuración, que son las personas que colaboran con la asesoría en el proyecto, trabajan con las capas 1, 2, 3 y 4 del modelo OSI (Open System Interconnection), para programar y configurar los servicios de CE e IPNG punto a punto y punto a multipunto en las diferentes plataformas con las que cuenta la compañía.

Para la configuración manual de los servicios mencionados (CE e IPNG), el ingeniero de configuración debe tener un amplio conocimiento sobre la topología de red, los protocolos L2-L3 que requiera el servicio y los comandos a utilizar dependiendo la plataforma en la que se va a configurar el servicio. Este proceso es algo dispendioso porque toma alrededor de 30 a 45 minutos la configuración de cada servicio, sin contar los posibles imprevistos a la hora del aprovisionamiento de los servicios, y teniendo en cuenta que la tasa de conexión de nuevos servicios tiene una tendencia al alza en la compañía. Por tal motivo, Internexa inició el desarrollo de un software en Python con el fin de automatizar la configuración de los servicios en lugar de realizar la configuración manual por medio de la interfaz CLI (Command Line Interface) en cada uno de los dispositivos de red.

En su momento, la primera versión de este software representaba una gran solución en cuanto a optimización en tiempo y recurso humano. Sin embargo, a mediados del año pasado dicho aplicativo pasó a ser obsoleto después una investigación conjunta con el proveedor de tecnología Huawei y los Ingenieros de Red de la compañía. Dicho estudio llegó a la conclusión de que había una mejor forma de configurar los servicios y mejorar el performance de los mismos. Por tanto, la forma en la que se venían configurando los servicios fue modificada y a partir de esa fecha se solicitó configurar los servicios de otra manera, razón por la cual habría que volver a ajustar el código de programación del software de configuración.

En este contexto, dado que el NOC actualmente no cuenta con recursos humanos que se dediquen específicamente a trabajar en el desarrollo de software dado el objetivo de negocio de la empresa, se optó por asignar un practicante dicha labor.

En la versión anterior anterior del software la configuración de los servicios de CE e IPNG se realizaba por medio de la asociación de VLANs (Virtual Local Area Network) a interfaces. En la actualidad, el software permite que los servicios sean entregados por medio de subinterfaces que contienen encapsulación del número de VLAN.

Esta nueva versión del software permite configurar de manera automática los servicios de CE e IPNG bajo la nueva arquitectura de configuración propuesta, implementa buenas prácticas de desarrollo en Python, y reduce los tiempos de gestión en un 80% frente a la configuración de servicios de manera manual. Esta reducción permite a los Ingenieros de Configuración realizar simultáneamente otras etapas correspondientes al proceso de configuración, mientras la canalización se realiza de manera automática.

2. Objetivos

Objetivo General

Desarrollar un software de gestión para la configuración automática de los servicios de Carrier Ethernet e IP Next Generation en la red MPLS Huawei de Internexa Colombia.

Objetivos Específicos

- Comprender la estructura y el funcionamiento de la red MPLS Huawei Colombia para el proceso y configuración de los servicios CE y IPNG.
- Reconocer las características del software con que dispone actualmente la organización.
- Modificar el software de gestión de acuerdo con los nuevos requerimientos de la organización.
- Validar el funcionamiento y entregar a la operación nuevo software para la configuración de servicios CE e IPNG.

3. Marco Teórico

3.1 Modelo OSI

El modelo OSI (Open Systems Interconnection) es un modelo conceptual creado por la ISO (Organización Internacional de Normas) con el fin de que diversos sistemas se puedan comunicar por medio de estándares y protocolos [1].

Los principios que se tuvieron en cuenta para crear el número de capas fueron: diferenciar el nivel de abstracción en cada capa, cada capa debe tener una función bien definida, las funciones de cada capa deben seguir unos protocolos estandarizados internacionalmente, y el número de capas debe ser el adecuado para que no haya funciones distintas en una misma capa [2].

En la Figura 1 se ilustra como el modelo OSI está compuesto por 7 capas que son:

1. Capa Física
2. Capa de Enlace
3. Capa de Red
4. Capa de Transporte
5. Capa de Sesión
6. Capa de Presentación
7. Capa de Aplicación

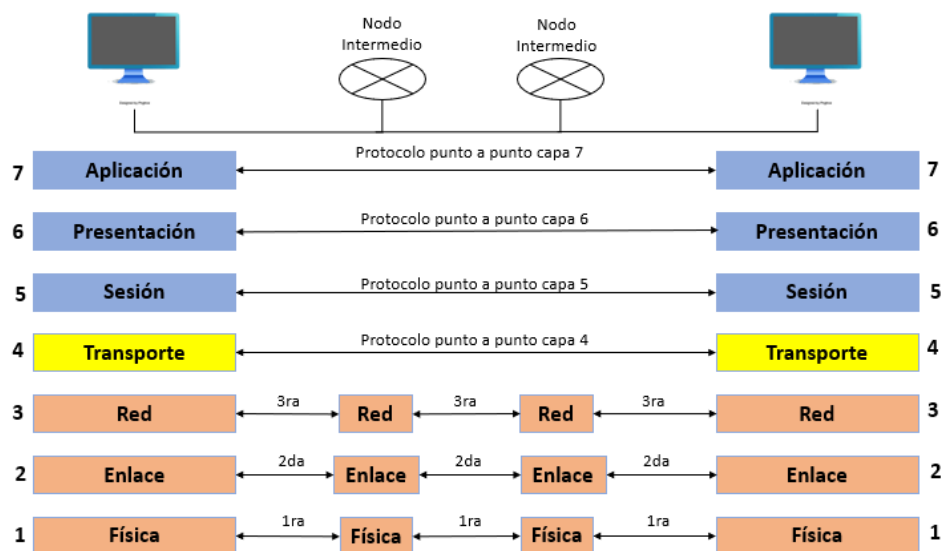


Figura 1: Estructura general del modelo OSI.

Los modelos de comunicación del Internet moderno no siguen estrictamente el modelo OSI pero aun así es de gran importancia debido a que si un usuario presenta problemas de conectividad, se puede dividir las diferentes instancias de una conexión y así encontrar el origen del problema [1].

En este proyecto se involucran diferentes capas del Modelo OSI. Por ejemplo, el software a desarrollar crea sesiones para gestionar dispositivos capa 2 y 3 (switches y enrutadores), usando protocolos capa 7 (SSH, Secure Shell). A su vez, los dispositivos de interconexión (capa 2 y 3) están desplegados a nivel nacional sobre una red de fibra óptica (capa 1).

3.1.1. Capa Física

Es la responsable de la transmisión de los bits a través del medio físico. Aspectos como la duración de un bit, la representación eléctrica de los bits 1's y 0's, la forma de establecer el inicio y el fin de las conexiones, y la simultaneidad de la comunicación son importantes para garantizar para que estos bits lleguen sin errores a su destino [2]. En esta instancia también están involucrados otros aspectos como los cables, conectores, entre otros (ver Figura 2).

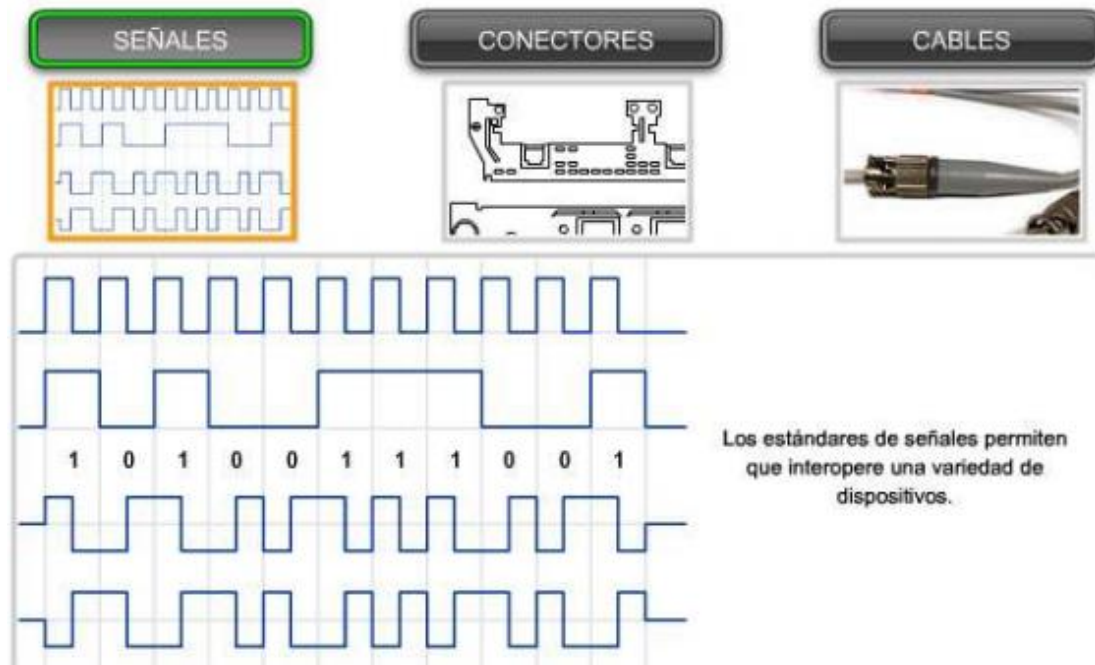


Figura 2: Elementos correspondientes a la capa física del modelo OSI.
Tomado de <https://blog.utp.edu.co/ee973/files/2012/04/capitulo08-Capa-Fisica.pdf>.

3.1.2. Capa de Enlace

Es la encargada de garantizar la comunicación entre dos dispositivos que se encuentran dentro la misma red. En esta capa se establece y se termina la conexión. Además, los paquetes se dividen en pequeñas piezas llamadas tramas, las cuales son enviadas por separado [1]. La trama está compuesta por dos partes las cuales son el LLC (Logical Link Control) donde se identifican los protocolos de red, se realiza la sincronización en el envío de las tramas y se controlan los errores. La otra parte es llamada MAC (Media Access Control) que usa las direcciones MAC para conectar los dispositivos y dar acceso a la transmisión y recepción de datos [3].

Al interior del desarrollo de este proyecto es clave comprender la estructura de la trama Ethernet, la cual contiene los siguientes campos: Dirección Destino, Dirección Origen, Tipo, Datos y Secuencia de verificación de la trama [1] (ver Figura 3).

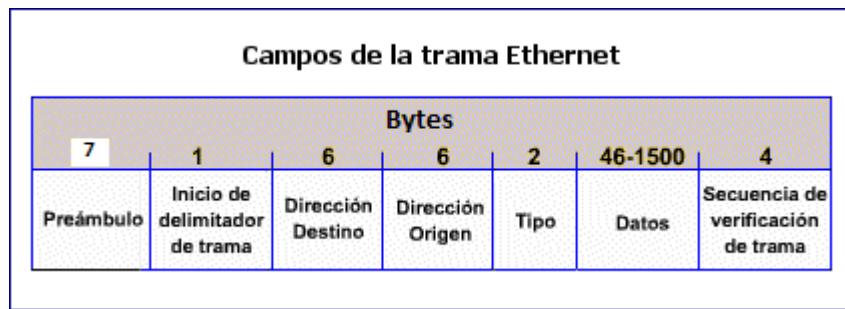


Figura 3: Campos contenidos en una trama Ethernet.

3.1.3. Capa de Red

Es la que se encarga de proporcionar conectividad entre dos dispositivos que se encuentran en distintas redes (ver Figura 4) [1]. Para garantizar lo anterior es necesario que se lleven a cabo dos procesos. En el primero similar a la capa de enlace, se dividen los datos en pequeñas piezas llamadas paquetes que se envían por separado para que luego en el destino se reensamblan. La otra función es descubrir la mejor ruta para llegar al destino, esto es conocido como enrutamiento [3].

Las rutas de los paquetes se encuentran en la tabla de enrutamiento almacenada en cada uno de los routers, que pueden ser estáticas o dinámicas que son las más usadas actualmente debido al crecimiento de las redes [2].

Manejar la congestión de los paquetes dentro de las subredes también es asunto de la capa de red para evitar los cuellos de botella. Desde una perspectiva más general se podría decir que en esta capa está relacionada con la Calidad del Servicio ya que se encuentran involucrados aspectos como retardo, tiempo de tránsito, entre otros [2].

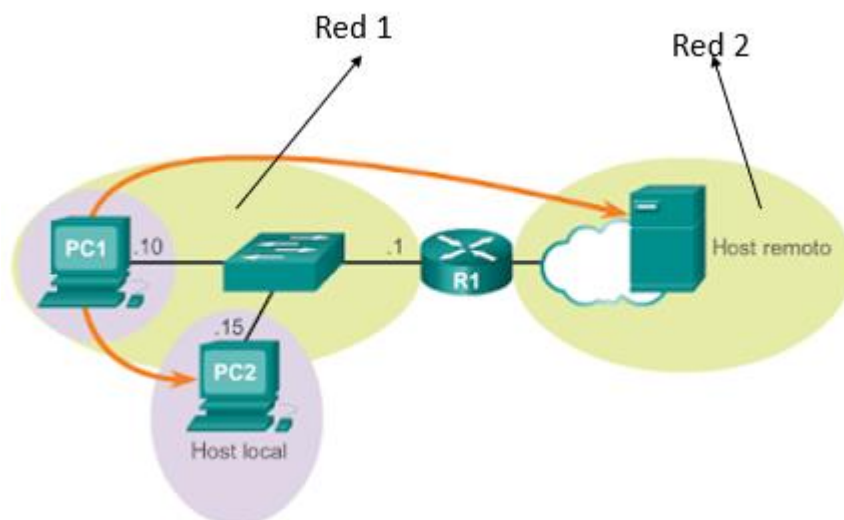


Figura 4: Funciones realizadas dentro de la capa de red.

3.1.4. Capa de Transporte

En esta capa se garantizan que los datos lleguen al destino correspondiente. Para esto debe tomar los datos de la capa de sesión y separarlos en pequeños pedazos llamados segmentos o datagramas antes de enviarlos a la capa de red. La capa de transporte del receptor lo que hace es reorganizar los segmentos o datagramas para que éstos pasen a la capa de sesión [1].

La capa de transporte en el receptor se encarga del control de los errores, asegurándose de que los datos lleguen completos y en el caso de que no sea así, se pedirá una retransmisión en algunos casos [1]. No siempre se pide retransmisión porque hay tipos de servicios en los que no hay garantía de la entrega correcta de los datos [2].

Esta capa a diferencia de las inferiores (Física, Enlace, Red), se realiza un control de flujo entre puntos extremos y no entre nodos intermedios (ver Figura 1). Además, establece el tipo de servicio que se debe proveer en la capa de sesión y a los usuarios de la red [2].

3.1.5. Capa de Sesión

Esta capa es responsable de abrir y cerrar los canales de comunicación entre dos dispositivos (ver Figura 5), dichos canales son llamados sesiones. La capa de sesión se asegura de mantener el canal de comunicación el tiempo necesario para que todos los datos sean transmitidos, y lo cierra para evitar el desaprovechamiento de los recursos de red [1].

En una sesión abierta están los servicios de control de dialogo, manejo de tokens y la sincronización en caso de posibles fallas [2].

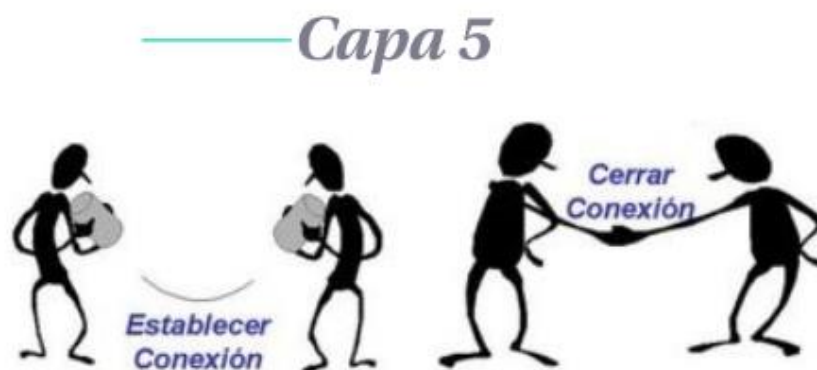


Figura 5: Funciones de la capa de sesión dentro del modelo OSI.
Tomado de https://prezi.com/qv8clvohyj_k/capa-de-sesion/

3.1.6. Capa de Presentación

La capa de presentación es la que se encarga de la traducción, cifrado, descifrado y compresión de los datos (ver Figura 6), para que estos sean consumidos dentro de la capa de aplicación [1].

A diferencia de las otras capas inferiores, en esta no existe la preocupación de transmitir 1's y 0's entre dispositivos, sino que se enfoca en la sintaxis y en la semántica de la información [2].



Figura 6: Funciones de la capa de presentación dentro del modelo OSI.
Tomado de <https://www.emaze.com/@ACFFQOCC/modelo-osi>.

3.1.7. Capa de Aplicación

Es la única capa con la cual el usuario interactúa dentro de la red. En esta capa están incluidos los protocolos utilizados en aplicaciones como Navegadores y clientes email con las cuales el usuario es donde inicia la aplicación. En esta capa están incluidos los protocolos como HTTP (HyperText Transfer Protocol) y SMTP (Simple Mail Transfer Protocol) [2].

Se tiende generalmente a confundir que el software desarrollado y su interfaz gráfica hacen parte de la capa de aplicación, pero no es así [1].

En la Figura 7 se muestran los protocolos más usados en los aplicativos de Internet.

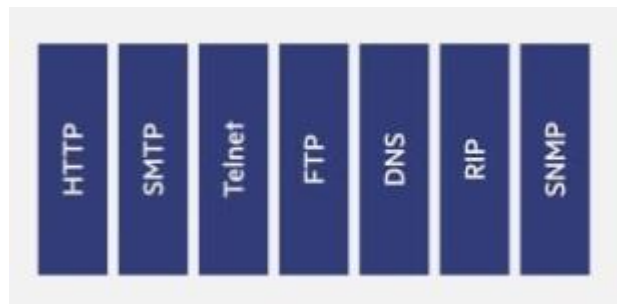


Figura 7: Protocolos de la capa 7 del Modelo OSI.
Tomado de [3].

3.2 Tecnología MPLS

La conmutación de etiquetas de múltiple protocolo MPLS (Multiprotocol Label Switching) es una técnica de transporte utilizada dentro de las redes WAN (Wide Area Network) de los proveedores para hacer más rápido el tráfico de datos ya que aprovecha la flexibilidad de los protocolos de capa 3 y la simplicidad de la conmutación (capa 2) [5].

Fue creado a finales de los años 90 por la IETF (Internet Engineering Task Force) debido a que para esa época el protocolo IP estaba dejando de ser eficiente con el crecimiento de las redes. El hecho de que cada enrutador intermedio tuviera que mirar su tabla de enrutamiento para decidir la ruta y el siguiente salto del paquete, ya lo convierte en un proceso super lento [6]. La solución que presenta MPLS para el

inconveniente anterior es que solo se analice el encabezado IP en los equipos de borde y no en los nodos de tránsito [5].

Aunque MPLS está basado en IPv4, también soporta otros protocolos de red como IPv6 e IPX (Internet Exchange Point). Además, se pueden transportar múltiples servicios que incluyen datos, voz multimedia y conexiones VPN (Virtual Private Network) [5].

La estructura de una red MPLS (ver Figura 8), está compuesta por los nodos LER (Label Edge Routers), los nodos de tránsito LSR (Label Switching Routers) y el LSP (Label Switched Path) que es el túnel por donde pasa el flujo de datos.

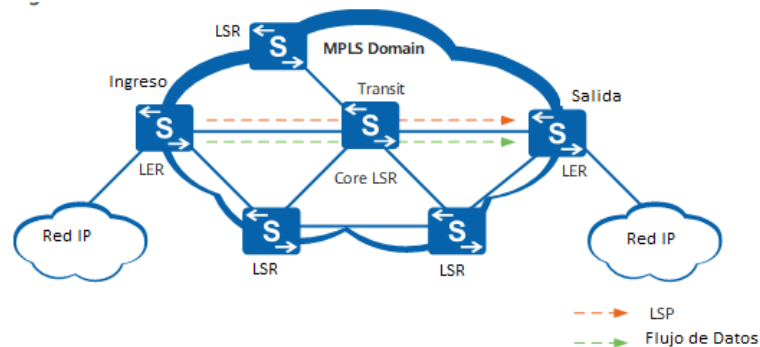


Figura 8: Estructura de una red MPLS.

El funcionamiento de MPLS consiste en que cuando un paquete entra a la red de backbone del proveedor por el router LER, inmediatamente es etiquetado. Este router decide que túnel toma el paquete para llegar a su destino. El encabezado MPLS se pone entre los encabezados de la capa 2 y 3 para que los routers de tránsito no miren la cabecera IP (ver Figura 9). Cada vez que un paquete pasa por un LSR, este realiza un cambio etiqueta y así sucesivamente con los otros hasta llegar al extremo del túnel. En el extremo del túnel se elimina la cabecera MPLS [7].

Una cabecera MPLS (ver Figura 9), tiene un tamaño de 32 bits que se distribuyen de la siguiente manera: 20 bits para la etiqueta MPLS, 3 bits para identificar la clase de servicio en el campo EXP, 1 bit que identifica la jerarquía de las etiquetas y 8 bits del TTL (Time to Live) [8].

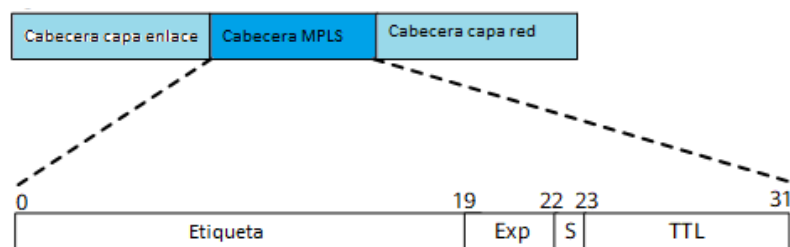


Figura 9: Cabecera MPLS dentro de un paquete.

3.3 Red MPLS Huawei Colombia

Para entregarle distintos servicios a sus clientes, Internexa cuenta con una red de core conocida como la Red MPLS Huawei Colombia que viene funcionando desde el 2011. Esta red cuenta con 6 anillos conformados por 37 nodos ubicados en sectores

estratégicos a nivel nacional, brindando una capacidad de comunicación de más de 100 Gbps (Gigabits por segundo).

Anteriormente los nombres de los equipos que integran esta red estaban basados en el código de aeropuertos IATA (International Air Transport Association), de acuerdo con la ciudad donde el dispositivo estuviera ubicado. Sin embargo, actualmente existen casos donde existen dos equipos por nodo y no todos son de la misma serie de Huawei. Por esta razón, la nueva nomenclatura consiste primero en colocar la serie del equipo (S12, S67 o S93), seguido del código IATA de la ciudad. En el caso de que haya 2 o más equipos de la misma serie en el mismo nodo, seguido del código IATA va una letra para distinguirlos (A, B...) [9].

La topología conformada por anillos creada por la empresa (ver Figura 10), proporciona redundancia y tolerancia a fallos ya que, con la caída de algún nodo principal, se podría redirigir el tráfico de datos por otro lado del anillo. Otra ventaja que ofrece esta topología es su escalabilidad, porque si se incluye un nuevo equipo a la red de core MPLS, no tendrían que desconectar los servicios de los clientes debido a que el flujo de datos pasaría por los otros equipos que conforman la red de anillo [9].

Los equipos Huawei que conforman la red MPLS de Huawei son switches de las series S12700, S6700 y S9300 [9].

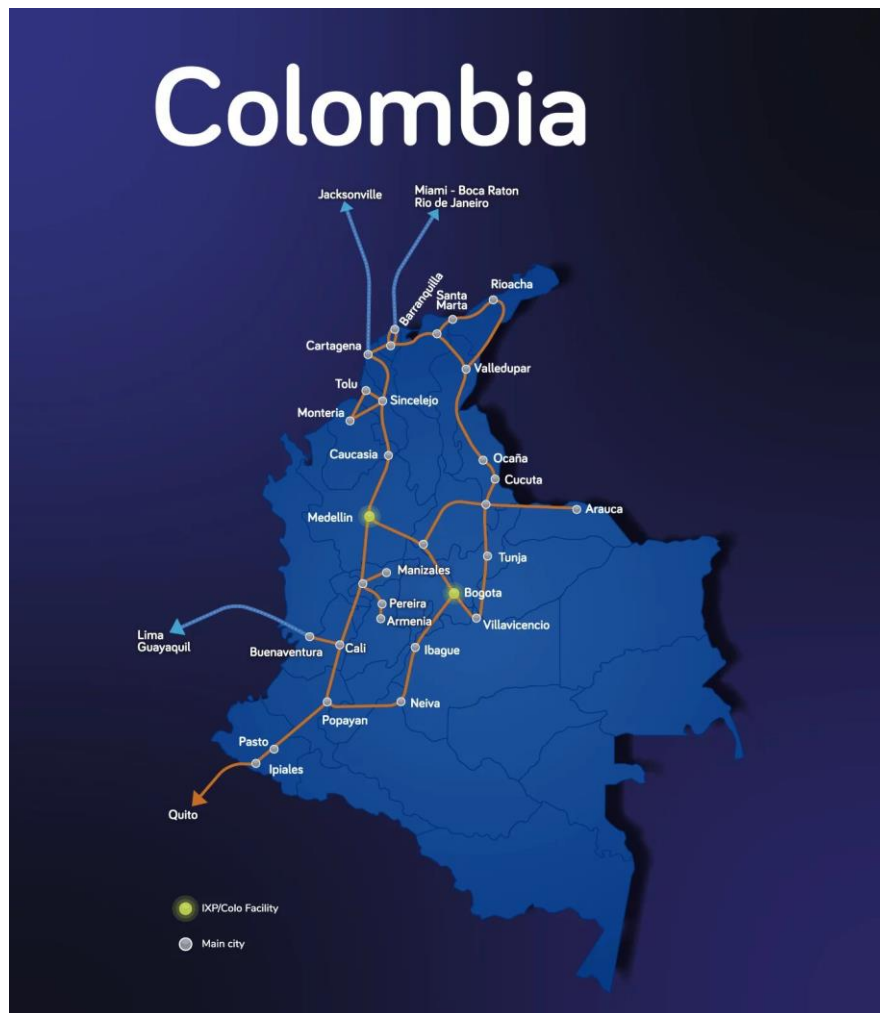


Figura 10: Topología de red en anillo de Internexa Colombia.
Tomado de <https://www.internexa.com/cobertura/cobertura/>

3.3.1. Switches Agile de la serie 12700 Huawei

Los switches de la serie 12700 son equipos de core diseñados para las redes de próxima generación. Proporcionan servicios capa 2 y capa 3 de alto rendimiento con garantía de más de 10 años, programable para el aprovisionamiento de nuevos servicios de manera flexible y rápida, y tienen convergencia para tecnologías cableadas e inalámbricas. En cuanto al hardware, esta serie de equipos cuenta con un millón de entradas MAC, 8 puertos de 100GE (Gigabit Ethernet) y 16 puertos de 40GE ideales para escenarios de redes MAN (Metropolitan Area Network) implementadas en instituciones de educación superior y grandes empresas [10].

Los aspectos más relevantes de esta serie de dispositivos a nivel de red son: soporte MPLS, IPv4, IPv6, 4000 VLANs, distintos tipos de interfaces (troncal, acceso, Híbrido y QinQ), protocolos de enrutamiento dinámico tanto para IPv4 como para IPv6, agregación de enlaces, Spanning Tree Protocol y QoS (Quality of Service) [11].

La serie S12700 cuenta con 3 modelos: 12704, S12708, 12712 (ver Figura 11), y cuentan con capacidad de conmutación y puertos expandibles. Su nuevo hardware adopta la ventilación left-to-rear que hace un buen aprovechamiento de la energía. Para

evitar el riesgo de una caída en los servicios gracias a que cuenta con una red de chips que almacenan energía [12].

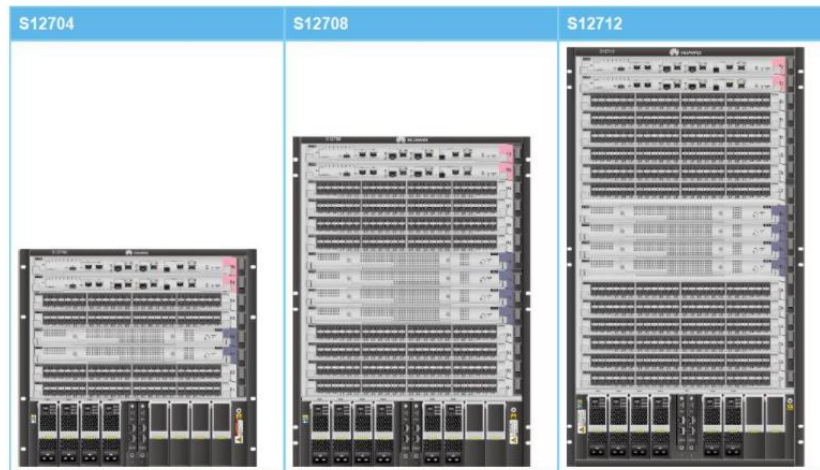


Figura 11: Switches de la línea 12700.
Tomado de [11].

3.3.2. Switches de enrutamiento S6700

La serie Huawei S6700 son unos switches de próxima generación que pueden ser utilizados en un campus universitario como en un DataCenter ya que proveen un ancho de banda de 10 Gbps. Además, cuenta con varios puertos ópticos 10GE [13].

Estos equipos proveen múltiples mecanismos de seguridad para prevenir distintos ataques informáticos como DoS (Denial of Service), falsos DHCP, suplantación de IP/MAC, entre otros. Los mecanismos que usa para prevenir las amenazas mencionadas anteriormente son autenticación por usuario, dirección IP, dirección MAC o VLAN ID. Otro aspecto que lo hace muy confiable en su funcionamiento es que posee alimentación eléctrica redundante [13].

Administrar su funcionamiento también es una cosa sencilla ya que puede ser configurado usando el CLI, SSH (Secure Shell) v2, Telnet, entre otros protocolos de acceso remoto [13].

Los aspectos más relevantes de las características de red es que cuenta con 4000 VLANs, asignación de VLAN por diferentes mecanismos, IPv4, IPv6, listas de control de acceso, QoS y protocolos de enrutamiento tanto para IPv4 como para IPv6. Además, soporta varias modalidades de MPLS lo cual es apropiado para los múltiples servicios que se prestan en la compañía, y tiene autenticación por AAA (Authentication, Authorization, Accounting), RADIUS (Remote Authentication Dial-In-User Service), y TACACS (Terminal Access Controller Access Control System) [13].

Algunos de los modelos son el S6700-24-EI, S6700-48-EI, S6720-30C y S6720-54C son ilustrados en la Figura 12.



Figura 12: Modelos de la serie S6700.
Tomado de [13].

3.3.3. Switches de enrutamiento S9300

Los switches de enrutamiento terabit de la serie Huawei S9300 son switches diseñados para redes multiservicio. El S9300 utiliza el concepto de conmutación inteligente multicapa de Huawei para proporcionar servicios de conmutación Capa 2 / Capa 3 de alto rendimiento [14].

Suministra aplicaciones de red enriquecidas como video de alta definición (HD - High Definition), computación en la nube elástica, hardware IPv6 seguridad unificada y H-QoS (Hierarchical Quality of Service – Jerarquía de Calidad de Servicio). Los switches S9300 se caracterizan por proporcionar funciones de enrutamiento y conmutación convergentes de extremo a extremo. Se utilizan en redes de área amplia, redes de área metropolitana y centros de datos para ayudar a los operadores a construir redes centradas en aplicaciones [14].

Entre las características de red más importantes se puede encontrar que cuentan con distintos tipos de interfaz (Troncal, Acceso, Híbrido y QinQ), conmutación por medio de VLAN, protocolos de enrutamiento dinámico tanto para IPv4 como para IPv6 y varios tipos de MPLS [14].

El S9300 está disponible en tres modelos: S9303, S9306 y S9312 (ver Figura 13). Todos los modelos S9300 usan módulos y componentes intercambiables para una capacidad de conmutación económica y expansión de puertos. Además, los modelos S9300 utilizan tecnologías innovadoras de ahorro de energía que reducen considerablemente el consumo de energía y el ruido sin comprometer el rendimiento o la estabilidad [14].



Figura 13: Modelos de la serie S9300.
Tomado de [14].

3.3.4. Servicio Carrier Ethernet

Un servicio Carrier Ethernet (CE) es un despliegue de tecnologías y servicios que surgieron a partir de la implementación de Ethernet dentro de las redes WAN a principios de los años 2000 [15].

La tecnología Ethernet era la preferida de los usuarios dentro del entorno de las LAN (Local Area Network), debido a su flexibilidad, formato universal de tramas y diseño sencillo. Por estos atributos es que se buscó la manera de ampliar Ethernet a las WAN, y resulto trayendo muchas ventajas ya que era mucho más económico y escalable que ATM (Asynchronous Transfer Mode) y Frame Relay [15].

Las ventajas que tiene CE son que cuenta con una gran variedad de usos (residencial, comercial, móvil) a grandes velocidades. Además, puede detectar y recuperarse de las fallas, métricas que le permiten prestar servicios de voz, video, datos y servicios móviles [16].

En Internexa el servicio CE se ofrece para conectar sedes de compañías a nivel de capa 2 del modelo OSI (ver Figura 14), es decir, que no cuenta con salida a internet. Estas conexiones pueden ser Punto a Punto o Punto a Multipunto, y el tráfico entre los canales puede ser de voz, datos, telefonía y Cloud [9].

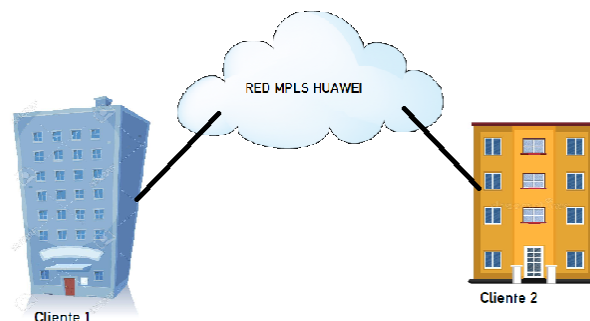


Figura 14: Ejemplo de servicio CE en Internexa.

3.3.5. IP Next Generation

Las Redes de Próxima Generación son una arquitectura basada e IP que permite unificar redes fijas y móviles. Estas redes soportan servicios como llamadas de voz, SMS (Short Message Service), Internet e IPTV (Televisión IP) [17].

Es capaz de explotar al máximo el ancho de banda del canal haciendo uso de las Tecnologías de Calidad del Servicio (QoS) de modo que el transporte sea totalmente independiente de la infraestructura de red utilizada. Además, ofrece acceso libre para usuarios de diferentes compañías y apoya la movilidad que permite acceso multipunto a los usuarios [17].

En INTERNEXA este servicio es conocido como IP Next Generation (ver Figura 15). Provee canal de internet y acceso a contenidos locales por medio de su CDN. En el software que se está realizando, este servicio solo se va a configurar punto a punto trabajando en las capas 2 y 3 del modelo OSI. Este servicio dentro de la empresa se conoce como un CE con salida a internet [9].

La compañía recibe internet por medio de los cables submarinos PCCS (Pacific Caribbean Cable System) y Sam-1 (South America 1) (ver Figura 16). Desde Barranquilla y Cartagena se provee internet a unos equipos Alcatel ubicados en ciudades como Cartagena, Bogotá, Cali y Medellín. Finalmente, estos equipos están conectados a uno de los nodos de la red MPLS [9].

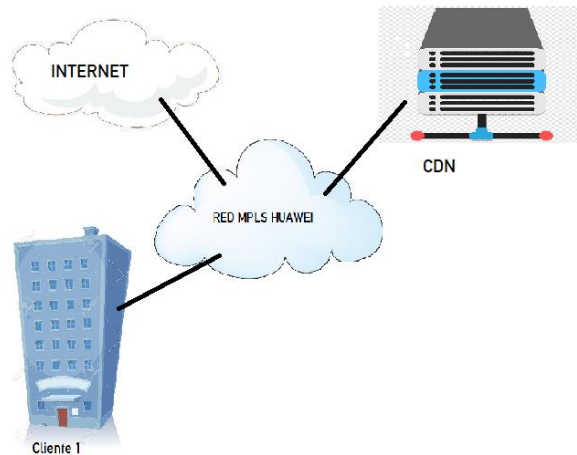


Figura 15 Ejemplo de un servicio IPNG para un cliente en Internexa.



Figura 16: Cables submarinos de Internexa Colombia.
Tomado de <https://www.submarinecablemap.com/>

3.4 VLAN

Una VLAN (Virtual Local Area Network) es una característica de comunicación de datos con la que cuentan algunos equipos, y es utilizada para dividir la LAN física en múltiples dominios de broadcast. Esta separación de dominios sirve para aislar los servicios con el objetivo de mejorar su gestión y seguridad [18].

Para el funcionamiento de una VLAN es necesario que el protocolo IEEE (The Institute of Electrical and Electronics Engineers) 802.1Q, ya que este se encarga de añadir una etiqueta de 4 bytes que se encuentra entre los campos de dirección MAC de origen

y tipo, en una trama Ethernet (ver Figura 17). Los switches tienen la capacidad de identificar paquetes provenientes de distintas VLAN [18].

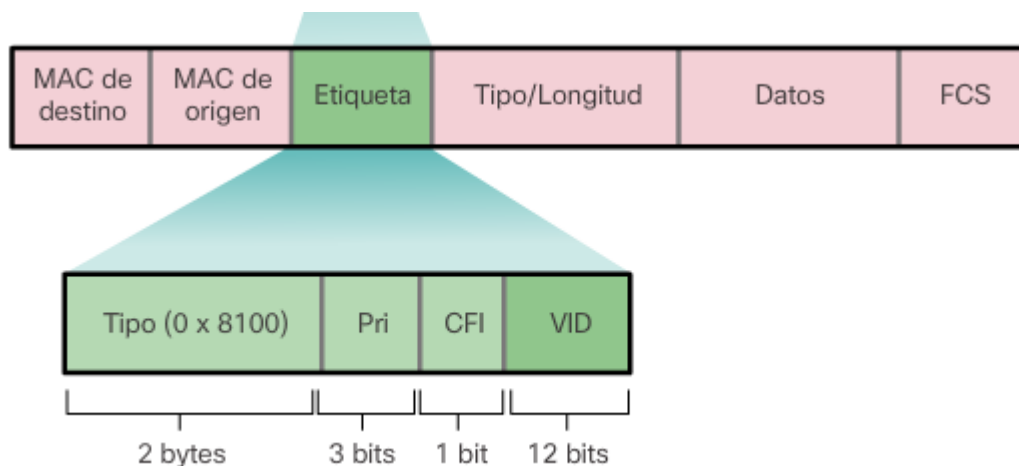


Figura 17: Etiqueta VLAN dentro de una trama Ethernet.

Tomado de <https://interpolados.wordpress.com/2017/05/01/etiquetado-de-tramas-de-ethernet-para-la-identificacion-de-vlan/>.

Como se puede ver en la Figura 17, existe un campo llamado VID (VLAN ID), que corresponde al número de VLAN y va desde el 0 hasta el 4095. Muchos dispositivos conectados al switch no son capaces de identificar las etiquetas de VLAN, por eso las interfaces deben configurarse para identificar las etiquetas y luego decidir si las agregan o las desagregan [18].

Un dispositivo que se encuentre conectado a una interfaz que pertenezca a un número de VLAN, solo se puede comunicar con otros dispositivos que tengan su mismo número de VLAN sin importar que estos se encuentren conectados a otro switch.

Anteriormente en la compañía, la configuración de los servicios para conectar a las compañías a nivel de capa 2 se hacía por medio de la asociación de números de VLAN a las interfaces de los equipos (ver Figura 18), ya que permitía separar el tráfico de las compañías. Sin embargo, ya no se utiliza esta configuración debido a que los clientes se quejaban constantemente con el funcionamiento de los servicios [9].

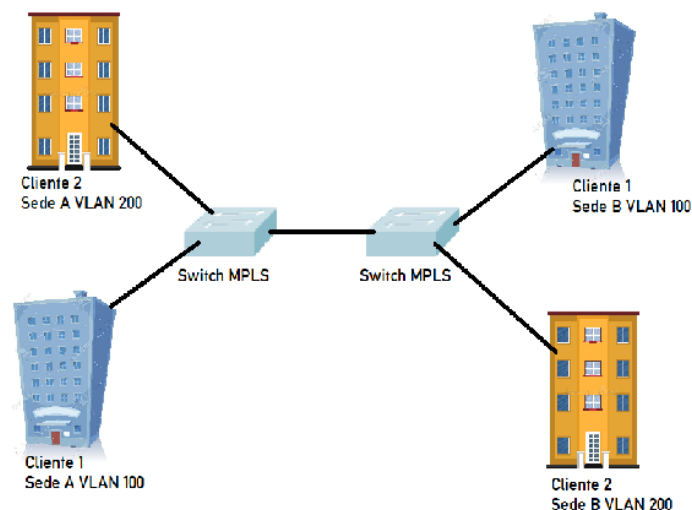


Figura 18: Ejemplo de clientes conectados por VLAN.

3.5 Subinterfaz

Las subinterfaces son un grupo de interfaces lógicas dentro de la misma interfaz física, las cuales permiten gestionar y asegurar de mejor manera el tráfico que pasa a través del medio físico [19].

Las subinterfaces representan una gran solución a la hora de entregar servicios ya que por la misma interfaz física pueden pasar muchas VLAN y se puede limitar el ancho de banda. También es menos costoso porque no sería necesario comprar más equipos de red [9].

Lastimosamente el principal inconveniente que representa las subinterfaces es que, si el puerto físico se cae, estas dejarían de funcionar [19].

Para el caso de los equipos Huawei, si se quiere configurar una subinterfaz, la interfaz principal debe estar en modo troncal para luego crear la subinterfaz y hacer la encapsulación del número de VLAN. Por ejemplo, si se quiere entregar un servicio con la VLAN 345 en el puerto GigabitEthernet 2/1/15, esta interfaz debe estar en modo troncal, la subinterfaz se llamaría GigabitEthernet 2/1/15.345 y luego hay que hacer la encapsulación dentro de la subinterfaz [9].

Después de recibir constantemente quejas por parte de los clientes al entregar servicios por interfaces VLAN, la compañía hizo un estudio conjunto con Huawei y llegaron a la conclusión que la mejor solución era utilizar subinterfaces [9].

3.6 Gestión de redes de Telecomunicaciones

La gestión de redes consiste en la planificación, control y organización de los elementos de comunicación con el objetivo de mejorar la disponibilidad, rendimiento y efectividad de los sistemas sobre un determinado costo [20].

Por medio de la gestión de red, se pueden establecer parámetros de calidad y control sobre cada uno de los servicios que se ofrecen a los usuarios y pasan a través de la red [20].

INTERNEXA cuenta con un NOC LATAM que se encarga de la supervisión y monitoreo de los servicios prestados 24/7 los 365 días del año. Esta área cuenta con especialistas en Carrier, Datos y Valor Agregado [9].

El software desarrollado en este proyecto se conecta a una base de datos del NOC, de la cual obtiene información como el listado de equipos, las direcciones IP, el historial de configuraciones realizadas, entre otros. Además, cuenta con una pestaña de agregar o eliminar equipos haciendo que la base de datos se actualice en tiempo real.

Los elementos que componen la gestión de red son los siguientes [20]:

- Agentes: Corresponde al software control que se encuentra en un equipo administrativo. Ahí está disponible la base de datos que contiene información de interés para el personal de red de una compañía.
- Gestores: Son las aplicaciones ejecutadas de manera periódica que controlan y supervisan todos los dispositivos permanentemente.
- Dispositivo administrativo: Corresponde a todos los dispositivos de red que se encuentran bajo la misma administración. De estos dispositivos se obtiene la información, por medio de protocolos como SSH.

3.7 Protocolo SSH

SSH (Secure Shell) es un protocolo que funciona bajo la arquitectura de cliente/servidor usado para establecer comunicaciones seguras con dispositivos remotos utilizando autenticación. Utiliza el puerto TCP 22 por defecto. A diferencia de Telnet o FTP (File Transfer Protocol), SSH si realiza un proceso de cifrado en el inicio de sesión (ver Figura 19) [21].

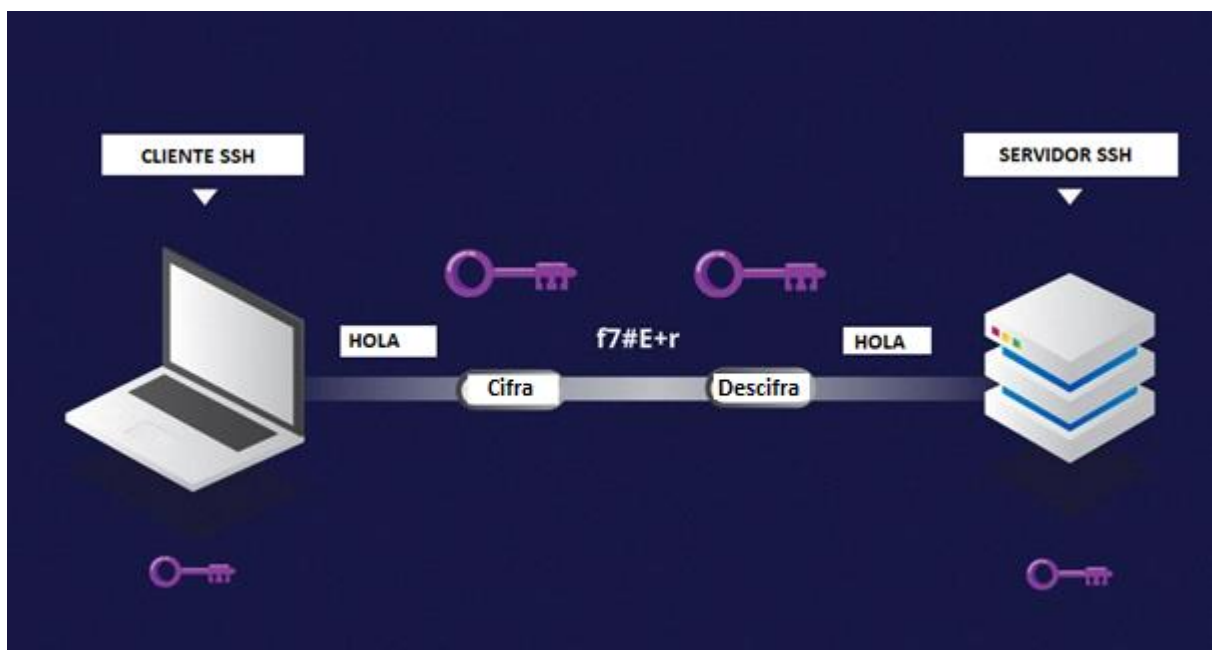


Figura 19: Proceso de cifrado en una conexión SSH.

El funcionamiento de SSH consiste en que el cliente es el que inicia la conexión con el servidor, luego existe una negociación entre los dos para definir cuáles serán los estándares de cifrado y finalmente el usuario se autentica [21].

En el software de configuración, las conexiones con los equipos de red se realizan usando SSH. Por políticas TI de la compañía, la autenticación por SSH utiliza TACACS, donde un servidor del NOC diferente al equipo de red es el que realiza la validación de las credenciales (ver Figura 20).

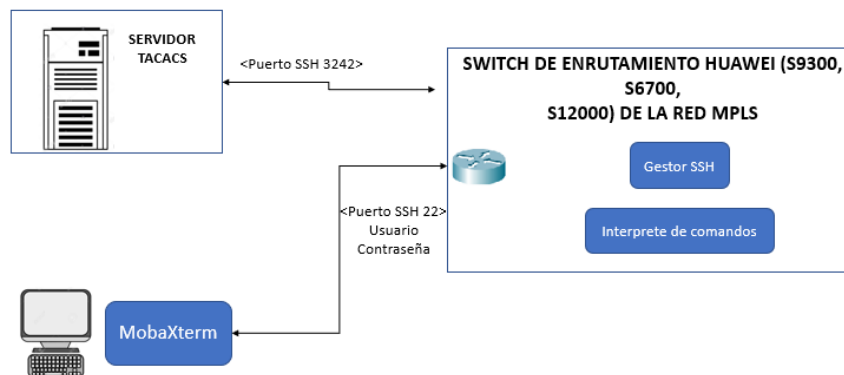


Figura 20: Esquema de autenticación remota usada en Internexa.

3.8 Descripción de Desarrollo de software

Durante el ciclo de vida de un sistema software, un gran número y variedad de partes interesadas intervienen de diversas maneras. Por esta razón se ve la necesidad de realizar una descripción del desarrollo de software clara y concisa con el fin de que los involucrados comprendan su funcionamiento. Existen muchas formas de representar los desarrollos de software de manera completa, la norma ISO/IEC/IEEE 42010 es una opción [22].

3.8.1. Norma ISO/IEC/IEEE 42010

El mundo de la ingeniería se ha visto en la necesidad de elaborar estándares de descripción software, de tal manera que cuando exista un desarrollo, sus autores puedan representar las relaciones entre los diferentes elementos que integran, los procesos que se gestan al interior del sistema, la manera en que el usuario interactúa con el sistema, cómo debe instalarse el sistema, etc. Un estándar como el 42010 permite que cualquier persona comprenda y tenga noción sobre el desarrollo planteado por alguien más, tal como se explica a continuación:

"La complejidad de los sistemas hechos por el hombre ha crecido a un nivel sin precedentes. Esto no solo ha creado nuevas oportunidades, sino que también ha traído mayores desafíos para las organizaciones que crean y utilizan sistemas. Los conceptos, principios y procedimientos de una arquitectura se aplican cada vez más para ayudar a administrar la complejidad con que se enfrentan las partes interesadas de los sistemas. La conceptualización de la arquitectura de un sistema, tal como se expresa en su descripción arquitectónica, facilita la comprensión de la esencia del sistema y de las propiedades claves relacionadas con su comportamiento, composición y evolución, que a su vez afecta aspectos tales como viabilidad, utilidad y mantenimiento del sistema. Las descripciones arquitectónicas son utilizadas

por las partes que crean, utilizan y administran sistemas modernos para mejorar la comunicación y la cooperación, lo que les permite trabajar de manera integrada y coherente.

Marcos de referencia arquitectónicos (architecture frameworks) y lenguajes de descripción de arquitecturas (ADLs, por sus siglas en inglés) se están creando como activos para codificar las convenciones y prácticas comunes del proceso arquitectónico y la descripción de arquitecturas dentro de diferentes comunidades y dominios de aplicación. Esta norma internacional aborda la creación, el análisis y el mantenimiento de arquitecturas de sistemas mediante la utilización de descripciones arquitectónicas. Esta norma internacional proporciona una ontología central para la descripción de arquitecturas. Las disposiciones de esta norma internacional sirven para imponer las propiedades deseadas en una descripción arquitectónica. Esta norma internacional también especifica disposiciones para imponer las propiedades deseadas de los marcos de referencia arquitectónicos y lenguajes de descripción de arquitecturas (ADLs), con el fin de apoyar de forma útil el desarrollo y utilización de descripciones arquitectónicas. Esta norma internacional proporciona una base sobre la cual comparar e integrar los marcos de referencia arquitectónicos y ADLs proporcionando una ontología común para especificar sus contenidos. Esta norma internacional se puede utilizar para establecer una práctica coherente para el desarrollo de descripciones arquitectónicas, marcos de referencia arquitectónica y lenguajes de descripción arquitectónica en el contexto de un ciclo de vida y sus procesos (no definidos por esta norma internacional). Esta norma internacional se puede usar adicionalmente para evaluar la conformidad de una descripción arquitectónica, de un marco de referencia arquitectónico, de un lenguaje de descripción de arquitectura, o desde un punto de vista arquitectónico (architecture viewpoints) según se disponga" [22].

En la Figura 21 se presenta los principales elementos de este estándar:

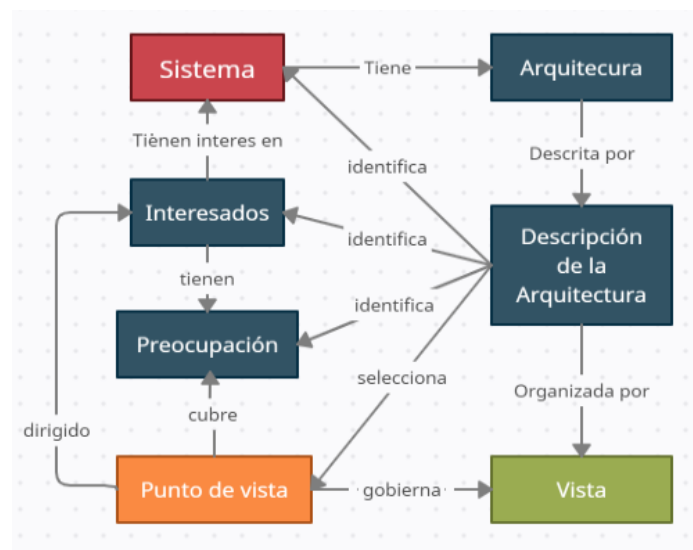


Figura 21: Representación Simple del estándar ISO/IEC/IEEE 42010.

De acuerdo con la Figura 21, para un Sistema determinado van a existir varios *Interesados*, y cada uno de los interesados va a tener una *Preocupación* con respecto al sistema. Así como cada interesado tiene una preocupación en específico, también tendrá un punto de vista de su preocupación con respecto al sistema [22]. Un punto de vista no es más que un conjunto de reglas para la construcción de vistas o representaciones de diferentes aspectos del software.

Además de lo anterior, cada sistema tendrá una respectiva *Arquitectura* y por consiguiente una *Descripción* de la arquitectura dada, que no es más que la recopilación de toda la documentación asociada a: el sistema, los interesados, las preocupaciones, los puntos de vista y las vistas como tal [22].

Para la representación del punto de vista se puede hacer uso de varios modelos, entre ellos destaca el “Modelo de Kruchten 4+1”, el cual proporciona una serie de vistas que permiten describir completamente la arquitectura de un desarrollo de software [23].

3.8.2. Modelo de Kruchten 4+1

El modelo de Kruchten 4+1, es un modelo de vistas diseñado por el profesor Philippe Kruchten y que se acopla con el estándar “ISO/IEC/IEEE 42010” que se utiliza para describir la arquitectura de un sistema software haciendo uso de múltiples puntos de vista [23].

Kruchten plantea un sistema de documentación de software de cinco vistas. Estas cinco vistas las denominó Kruchten como: vista lógica, vista de procesos, vista de despliegue, vista física y la vista “+1”. Esta última vista tiene la función de relacionar las 4 primeras vistas citadas, y la denominó vista de escenario (ver Figura 22) [23].

Cada una de estas vistas describe toda arquitectura del software que se esté documentando, pero cada una de ellas se documenta de forma diferente y muestra aspectos diferentes del sistema. A continuación, se explica que información debe contener la documentación de cada una de estas vistas [23].



Figura 22: Vistas de Kruchten.

Tomado de <https://jarroba.com/modelo-41-vistas-de-kruchten-para-dummies/>

3.8.2.1. Vista Lógica

Esta vista representa la distribución del sistema y lo que éste debe hacer. Aquí se presentan las funciones y/o clases implementadas en el desarrollo de software.

Para completar la documentación de esta vista se pueden incluir los diagramas de clases, de comunicación o de secuencia de UML (Unified Modeling Language) [23].

3.8.2.2. Vista de Despliegue

En esta vista se presenta el sistema desde la perspectiva de un programador o desarrollador. En otras palabras, se va a mostrar cómo está dividido el sistema software en componentes o módulos y las dependencias que hay entre éstos.

Para completar la documentación de esta vista se pueden incluir los diagramas de componentes y de paquetes de UML [23].

3.8.2.3. Vista de Procesos

En esta vista se muestran detalladamente los procesos que hay en el sistema y la forma en la que se comunican entre ellos. Se representa desde la perspectiva de un integrador de sistemas, el flujo de trabajo paso a paso y las operaciones de los componentes que conforman el sistema. Para completar la documentación de esta vista se puede incluir el diagrama de actividad de UML o un diagrama de flujo [23].

3.8.2.4. Vista Física

En esta vista se muestra, desde la perspectiva de un ingeniero de redes, todos los componentes físicos del sistema, así como las conexiones físicas y lógicas entre los componentes que conforman la solución (incluyendo los servicios). Se deben detallar también los protocolos de conexión de las componentes físicas. Para completar la documentación de esta vista se puede incluir el diagrama de despliegue de UML [23].

3.8.2.5. “+1” Vista de Escenarios

Esta vista representa la funcionalidad que el sistema software proporcionará a los usuarios finales. Para completar la documentación de esta vista se pueden incluir el diagrama de casos de uso de UML [23].

4. Metodología

El desarrollo de la práctica se divide en 4 etapas fundamentales donde cada una de ellas tiene diferentes tipos de actividades. Con esta división se garantiza el cumplimiento de los objetivos de manera eficiente y específica.

Primero se hizo un estudio de conceptos básicos de redes, así como de la tecnología MPLS dentro de una red WAN y los servicios CE e IPNG que ofrece la compañía. También se abordaron los dispositivos de red y se crearon sesiones remotas para conocer la configuración de los servicios CE e IPNG. Todas estas tareas fueron realizadas bajo el apoyo y supervisión del personal del NOC.

La segunda parte consistió en aprender comandos básicos de Huawei, comprender los tipos de servicios ofrecidos por la compañía y de enlaces que existen en los equipos. Después de tener claro lo anterior, los Ingenieros de Configuración realizaron la entrega de la primera versión del software para entender el código.

Con el código a disposición, se realizaron las modificaciones necesarias para cumplir con los nuevos lineamientos de configuración propuestos por la compañía, utilizando las recomendaciones de buenas prácticas para escritura de código en Python. Durante este proceso, cada modificación relevante realizada al software fue

revisada por los Ingenieros de Configuración, de modo de tomar cualquier nuevo requerimiento en la fase de desarrollo.

Por último se realizaban pruebas del software, actualización de los manuales, revisión de los tiempos de configuración y entrega formal del software a la compañía.

4.1 Estudio de conceptos básicos

Las primeras semanas dentro de la compañía, aparte del proceso de inducción, consistieron en reuniones con los Ingenieros de Configuración en las que explicaron cómo está compuesta a nivel de capa física la red de core de la compañía, las distintas formas de conectar equipos por medio de fibra óptica, la importancia de los cables submarinos para el aprovisionamiento de internet, los diferentes servicios ofrecidos a los clientes y la segmentación que se realiza para gestionar las redes en cada país. También se definió el plan de trabajo y las primeras tareas a realizar. Aparte de las reuniones, también se realizó un repaso de los conceptos básicos como conmutación, enrutamiento y subnetting. También se hizo la lectura sobre MPLS basado en la documentación de Huawei, entendiendo el uso de etiquetas y que ofrece muchas más ventajas que el enrutamiento por IP en cuanto a velocidad y capacidad de datos.

Después de los estudios previos ya los configuradores explicaron el funcionamiento de la red MPLS dentro de la compañía por medio de dibujos y visualizaciones en los equipos. Con estas sesiones se contextualizaron los conocimientos básicos de redes ya que para implementar MPLS, primero se debe aplicar un direccionamiento IP y un protocolo de enrutamiento IGP (Internal Gateway Protocol) para que los equipos de la red WAN se comuniquen entre ellos. Además, se aprendió nuevos conceptos como túneles, IP de gestión, IP de loopback, el nombre específico de cada equipo, el nombre alterno y los tipos de interfaces.

Durante esta etapa, el Ingeniero de Modelo de Operación hizo los trámites para que el SOC (Security Operation Center) creara una cuenta para la VPN del NOC. También hizo la asignación de una Máquina Virtual alojada en un servidor del NOC y una cuenta en la que podía realizar mis tareas ya que la laptop asignada por la empresa no contaba con esos permisos.

Ya teniendo la Máquina Virtual, se procedió a crear sesiones con los equipos de la red MPLS por medio de SSH a través del puerto 22 usando el aplicativo MobaXterm con una cuenta que solo tenía permisos para comandos tipos "*display*". Para que los equipos puedan identificar que permisos tiene asignado cada usuario, el equipo se conecta a un servidor TACACS por medio SSH a través del puerto 3242 (ver Figura 20).

TACACS es un protocolo de acceso que permite a un equipo remoto conectarse a un servidor para determinar si las credenciales son correctas y con qué permisos cuenta el usuario para interactuar con el equipo. La principal ventaja que ofrece es que se basa en AAA donde almacena los siguientes datos:

- Usuario que se conecto
- Hora a la que hizo la conexión
- Que comandos ejecutó el usuario.

4.2 Contextualización en el aprovisionamiento de servicios CE e IPNG.

Después de tener claro el concepto de MPLS dentro de la compañía, se hicieron sesiones con los Ingenieros de configuración donde ellos hicieron la explicación de que es un servicio CE y un IPNG, que son los abarcados dentro del proyecto. En la explicación ellos dijeron que entre los dos había muchas similitudes porque el tipo de tráfico (voz, datos, multimedia) es el mismo y que ambos emplean la red MPLS, pero en CE los extremos son equipos capa 2 mientras que en IPNG en uno de los extremos hay un equipo router de marca Alcatel.

Después de eso, ellos explicaron los tipos de puerto que se pueden configurar en los equipos, los cuales son Troncal, Acceso y QinQ. En los puertos troncales se necesita un tag de VLAN para que pasen varios servicios por ahí y son el caso de configuración más común. Los puertos de acceso no requieren configuración de VLAN y se entregan cuando el cliente desea un acceso exclusivo. Los QinQ son los menos comunes y solo se configuran cuando se va a entregar un servicio L2VPN.

Luego de conocer bien los servicios, los configuradores que utilizaron los equipos Huawei mostraron cuales son los datos necesarios para configurar un servicio, dependiendo de éstos variará la configuración. Para que los Ingenieros de Configuración sepan que datos utilizar, primero los clientes se comunican con los Ingenieros comerciales para organizar los requerimientos. Luego estos requerimientos pasan al Ingeniero de Servicios quien es el que asigna el número de VLAN y el ID del servicio. Finalmente, toda la información se pasa a los Ingenieros de Configuración que son los que se encargan de la canalización. En la Figura 23 se muestran cuáles son los datos necesarios para configurar cualquier servicio.

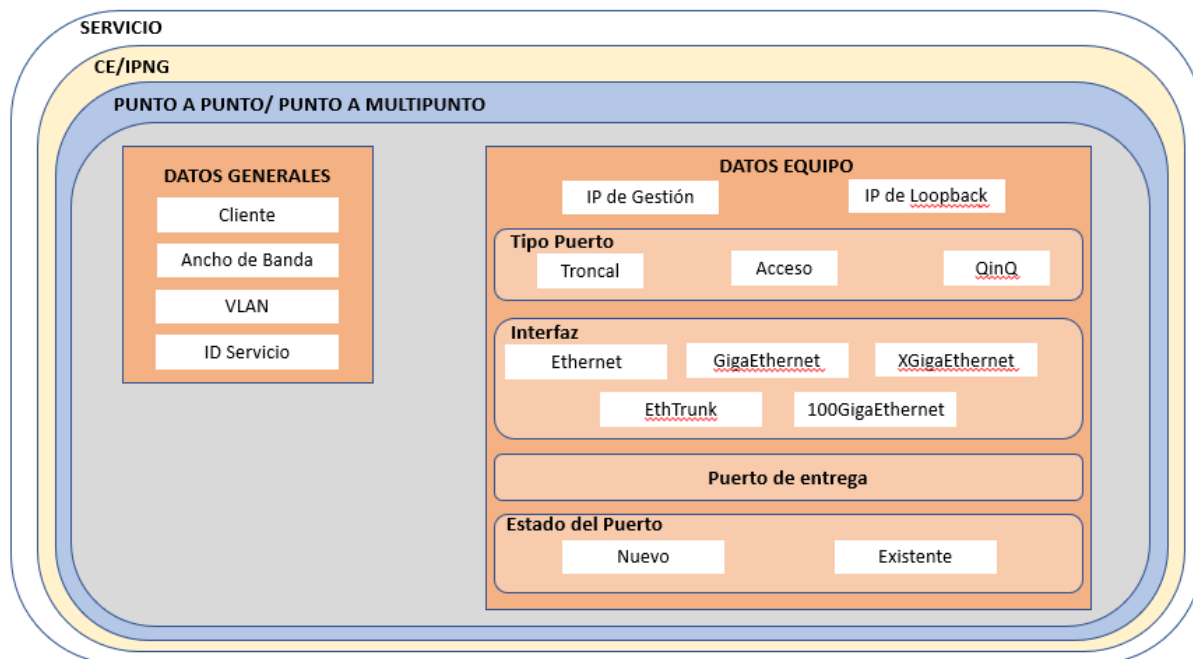


Figura 23: Datos esenciales para configurar cualquier servicio.

Dentro de la empresa se configuran dos tipos de enlace los cuales son Punto a Punto y Punto a Multipunto. El enlace Punto a Punto o VLL (Virtual Leased Lines) consiste en la conexión entre dos equipos. El enlace Punto a Multipunto o VSI (Virtual Switch

Instance) consiste en la conexión entre varios equipos (desde 3 hasta 6 equipos), usando topología estrella o full mesh. Ya con el conocimiento de los datos necesarios para canalizar cualquier servicio, los Ingenieros de Configuración mostraron cual es la diferencia de configurar un VLL y un VSI dentro de los equipos Huawei.

En el programa de acuerdo con el tipo de servicio, si se va a configurar un servicio IPNG aparecerán otros campos adicionales correspondiente al equipo Alcatel. En cuanto al tipo de enlace, si se escoge un punto a multipunto aparecerán otras opciones.

4.2.1. Datos Generales

El primer dato corresponde al **Cliente**, empresa o compañía que adquiere el servicio. Este nombre va a ser utilizado más adelante, durante la ejecución de comandos de configuración para la marcación las subinterfaces y los puertos de entrega del servicio. El segundo dato corresponde al **Ancho de Banda** que en Internexa es ofrecido desde 2 Mbps (Megabits por segundo) hasta 100 Gbps (Gigabits por segundo). El tercero es el número de **VLAN**, que sirve para crear la encapsulación de las subinterfaces, este va desde 2 hasta 4095. Y por último está el **ID Servicio**, corresponde al número que va a identificar el servicio, este número es único, exclusivo e irrepetible, solamente se podrá volver a utilizar siempre y cuando el servicio que tenía asociado haya sido cancelado.

4.2.2. Datos del equipo

Dentro de los datos del equipo se encuentra la **IP de gestión**, la cual permite realizar la conexión SSH con el equipo, para realizar su posterior configuración o monitoreo. La **IP loopback** permite localizar los túneles de comunicación entre este equipo y el resto de los equipos de la red MPLS. Actualmente existen 3 tipos de configuración de puerto, los cuales son *Troncal*, *Acceso* y *QinQ*.

InterNexa tiene a la disposición de sus clientes 5 tipos de interfaces de enlaces físicos, la utilización de cada una de estas viene definida por el ancho de banda del servicio. Se utilizará la interfaz **Ethernet** para aquellos servicios que cuenten con un ancho de banda en un rango que va desde 2 Mbps hasta 99 Mbps. La interfaz **GigaEthernet** se utilizará para aquellos servicios que cuenten con un ancho de banda en un rango que va desde 100 Mbps hasta 999 Mbps. La interfaz **XGigaEthernet** se utilizará para aquellos servicios que cuenten con un ancho de banda en un rango que va desde 1 Gbps hasta 10 Gbps. La interfaz **100GigaEthernet** se utilizará para aquellos servicios que cuenten con un ancho de banda de 100 Gbps. Por otra parte, la interfaz **Eth-Trunk** es un enlace lógico constituido a partir de los enlaces físicos vistos anteriormente. El ancho de banda que pasa a través de este enlace será la sumatoria del ancho de banda existente en los enlaces físicos asociados al servicio.

Finalmente, a un puerto se le pueden asociar dos estados. **Nuevo**, este estado implica que cuando se vaya a entregar un servicio en dicho puerto se requiere la ejecución de comandos adicionales para su creación y configuración. **Existente**, estado que no requiere la ejecución de órdenes para la creación de un puerto.

Con el conocimiento de los datos adquiridos, los Ingenieros de Configuración procedieron a mostrar en los equipos como es que se hacían antes las configuraciones por medio de interfaz VLAN y cuál es la nueva forma de configuración para que los servicios se propaguen por subinterfaz. Para comprender de mejor manera todos los conceptos y aspectos explicados fue necesario aprender comandos básicos de Huawei con los cuales se visualizan las configuraciones generales, los túneles creados por los especialistas de red, las marcaciones de cada interfaz, el estado de cada puerto y la configuración de cada servicio.

Teniendo claro lo anterior, finalmente hicieron entrega del código que correspondía a la primera versión del software, que estaba elaborado con el lenguaje de programación de Python.

4.3 Construcción de la nueva versión del software

Para poder empezar a desarrollar la segunda versión del software, fue necesario como primera medida entender como era el funcionamiento de la anterior versión con la que contaba la empresa.

Luego de conocer el funcionamiento general del software, se procedió a realizar las modificaciones en el código para cumplir con los nuevos requerimientos de la compañía.

Finalmente se realizaban pruebas por etapas con el acompañamiento de los Ingenieros de configuración, donde ellos daban su aprobación, los pasos a seguir en la modificación del código y retroalimentaciones.

4.3.1. Estudio del funcionamiento de la primera versión del software

Para lograr entender el funcionamiento de la primera versión del Software, primero se procedió a leer el manual de configuración y el manual de usuario. El manual de configuración es un documento que contiene las principales funciones del programa documentadas con la convención PEP (Python Enhancement Proposal) 257. También contiene instrucciones para acceder a las bases de datos ubicadas en el programa y la explicación de la convención PEP 257.

El manual de usuario es otro documento que contiene las secciones del software, la configuración básica de un servicio, como añadir o eliminar un equipo de la base de datos, como añadir o eliminar una troncal protegida, los mensajes de advertencia que aparecen durante la ejecución del programa y los requerimientos de software para su funcionamiento.

Luego de haber leído los manuales, se procedió a documentar el paso a paso del algoritmo, teniendo en cuenta el orden de aparición de las funciones.

A continuación, se describirá cómo funcionaba de manera resumida la primera versión del software. Adicionalmente, en cada ítem se mencionará las librerías que utilizan en caso de ser necesario.

4.3.1.1. Carga de información desde la base de datos

Cuando el programa se ejecuta, inmediatamente carga la información necesaria de una Base de Datos (BD) alojada en un servidor del NOC de Internexa. En esta BD se encuentran 5 tablas:

- La primera contiene información de los equipos Huawei de la red MPLS de Internexa y está integrada por los siguientes campos: Indice, Nombre, IP_Gestión, IP_Loopback, Nombre_Alternativo y Nombre_Real (ver Figura 24).

| Indice | Nombre | IP_Gestion | IP_Loopback | Nombre_Alternativo | Nombre_Real |
|--------|--------|------------|-------------|--------------------|-------------|
| 1 | TLU | | | TLU | S93aTLUITX |
| 2 | TUN | | | TUN | S93aTUNITX |
| 3 | NVA | | | NVA | S93aNVAEBO |
| 4 | CLO | | | CLO | S93aCLOCHI |
| 5 | LVI | | | LVI | S93aLVIISA |
| 6 | IBE | | | IBE | S93aIBEISA |
| 7 | BAQ | | | BAQ | S93aBAQNOG |
| 8 | BUN | | | BUN | S93aBUNPTC |
| 9 | RCH | | | RCH | S93aRCHITX |
| 10 | CUC | | | CUC | S93aCUCSMA |
| 11 | CTG | | | CTG | S93aCTGCHM |
| 12 | EPL | | | BOG | S93aBOGEPL |
| 13 | C100 | | | BOG | S93aBOGC100 |
| 14 | TCO | | | BOG | S93aBOGTCO |
| 16 | SMR | | | SMR | S93aSMRBCH |
| 17 | W10 | | | BOG | S93aBOGW10 |
| 18 | PPN | | | PPN | S93aPPNSBN |
| 19 | DCF | | | MDE | S93aMDEDCF |
| 20 | SHE | | | MDE | S93aMDESHE |
| 21 | ZFTA | | | ZFT | S93aBOGZFT |
| 22 | ZFTB | | | ZFT | S93bBOGZFT |
| 23 | VUP | | | VUP | S93aVUPECA |
| 24 | PEI | | | PEI | S93aPEIEAC |
| 25 | SIN | | | SIN | S93a5INBDB |
| 26 | BGA | | | BGA | S93aBGAPDP |
| 27 | PSO | | | PSO | S93aPSOJAM |
| 28 | VVC | | | VVC | S93aVVCITX |
| 29 | FUN | | | FUN | S93aFUNSTC |
| 30 | MTR | | | MTR | S93aMTRMAL |
| 31 | MDEA | | | MDE | S93aMDECGT |
| 32 | BJA | | | BJA | S93aBJASHE |
| 33 | DCT | | | BOG | S93aBOGDCT |
| 34 | ECO | | | ECO | S93aECOSTC |
| 35 | WSOB | | | BOG | S93bBOGWSO |

Figura 24: Tabla que contiene información de los equipos Huawei.

- La segunda tabla contiene información relacionada con los equipos Alcatel y está integrada por los campos: Indice, Nombre, IP_Gestión, Nombre_Alternativo y Nombre_Real (ver Figura 25).

| Indice | Nombre | IP_Gestion | Nombre_Alternativo | Nombre_Real |
|--------|--------|------------|--------------------|-------------|
| 1 | BOGWSO | | BOG | A77aBOGWSO |
| 2 | BOGZFT | | BOG | A77aBOGZFT |
| 3 | CLOCHI | | CLO | A77aCLOCHI |
| 4 | CTGCHM | | CTG | A77aCTGCHM |
| 5 | MDEA | | MDE | A77aMDECGT |
| 6 | MDEB | | MDE | A77bMDECGT |

Figura 25: Tabla que contiene información de los equipos Alcatel.

- La tercera tabla contiene la información de los equipos protegidos Alcatel, y está integrada por los campos: Indice, Nombre_Real, Nombre, Interfaz, Lag_Puerto y Descripción (ver Figura 26). De los campos mencionados, el campo lag_puerto corresponde a un puerto físico del equipo tipo Ethernet o a un enlace agregado que está conformado por varios puertos.

| Indice | Nombre_Real | Nombre | Interfaz | Lag_Puerto | Descripcion |
|--------|-------------|--------|----------|------------|------------------------------------|
| 1 | A77aCLOCHI | CLO | | | Descripcion temporal CLO-MDEB IPNG |
| 5 | A77aCTGCHM | CTGCHM | | | Descripcion de ejemplo |

Figura 26: Equipos protegidos Alcatel.

- La cuarta tabla contiene la información de los equipos protegidos Huawei, y está integrada por los campos: Indice, Nombre_Real, Nombre, Interfaz, Lag_Puerto y Descripción (ver Figura 27)

| Indice | Nombre_Real | Nombre | Interfaz | Lag_Puerto | Descripcion |
|--------|-------------|--------|----------|------------|-----------------------------------|
| 1 | S93aMDECGT | MDEA | | | Descripcion temporal MDEA-MDEB CE |
| 3 | S93aCLOCHI | CLO | | | Descripcion temporal CLO-BUN CE |
| 10 | S93aCTGCHM | CTG | | | Otra descripción |
| 14 | S93aCUCSMA | CUC | | | descripcion |

Figura 27: Equipos protegidos Huawei.

- La quinta tabla contiene el historial de configuraciones realizadas por el software, y está integrada por los campos: Índice, Fecha, Usuario, Tiempo y Descripción (ver Figura 28)

| Indice | Fecha | Usuario | Tiempo | Descripcion |
|--------|---------------------|----------|---------|---|
| 356 | 2021-05-19 13:50:07 | jmontoya | 1' 36'' | RINKU [37517] S93bMDECGT_S93aMDESHE_INTE_CO |
| 357 | 2021-05-20 09:20:49 | jmontoya | 1' 33'' | ESU [36871] S67aMDEGTD_S93bMDECGT_INTE_CO |
| 358 | 2021-05-20 10:33:48 | caperez | 3' 9'' | SELTIC [37446] S93aCTGCHM_S93aSMRBCH_INTE_CO |
| 359 | 2021-05-20 18:47:59 | caperez | 1' 11'' | CONCRETO [36681] S93aMDECGT_S93aMDESHE_ETHE_CO |
| 360 | 2021-05-21 08:12:02 | caperez | 1' 11'' | CONCRETO [36681] S93aMDECGT_S93aMDESHE_ETHE_CO |
| 361 | 2021-05-21 11:01:57 | caperez | 1' 40'' | NEUTRONA [36927] S93aB0GWS0_S93aB0GDC ETHE_CO |
| 362 | 2021-05-21 15:07:20 | jmontoya | 3' 26'' | INTEGRA_MULTI [37437] S93aB0GWS0_S93aIBEISA_INTE_CO |
| 363 | 2021-05-21 17:29:32 | jmontoya | 2' 37'' | RRORRO [12345] S93aVUPECA_S93aIBEISA_ETHE_CO |
| 364 | 2021-05-24 15:49:38 | caperez | 1' 39'' | COLOMBIA_TEL [36734] S93aMDECGT_S93aBAQNOG_ETHE_CO |

Figura 28: Historial de configuraciones.

Para importar toda esta información se utiliza la librería de MySQL, que es un servicio de BD basado en la arquitectura cliente servidor, la cual permite a los desarrolladores de manera rápida y segura el desarrollo de aplicaciones en la nube. La principal ventaja que ofrece es que, al ser código abierto permite usar su código fuente sin restricciones en cualquier programa soportando librerías, herramientas administrativas y APIs (Application Programming Interface) [24].

4.3.1.2. Interfaz gráfica de Usuario (GUI)

La primera versión del software está compuesta por 5 pestañas, donde en cada una hay una funcionalidad diferente. En la Figura 29 se puede ver cuáles son esas pestañas.

| | | | | |
|-----------|---------------|-------|-----------|----------------------|
| Principal | Configuración | Ayuda | Historial | Troncales Protegidas |
|-----------|---------------|-------|-----------|----------------------|

Figura 29: Pestañas de la primera versión del software.

La pestaña **Principal** de la primera versión cumplía con los requerimientos del NOC, ya que contenía un área para ingresar la información (ENTRADAS), un área para visualizar los resultados (SALIDAS) y otra en la que están contenidas los botones

(BOTONES). En la Figura 30 se puede ver cómo era la estructura de la pestaña **Principal** en su primera versión y en la Figura 31 se muestra la implementación de dicha Interfaz Gráfica.

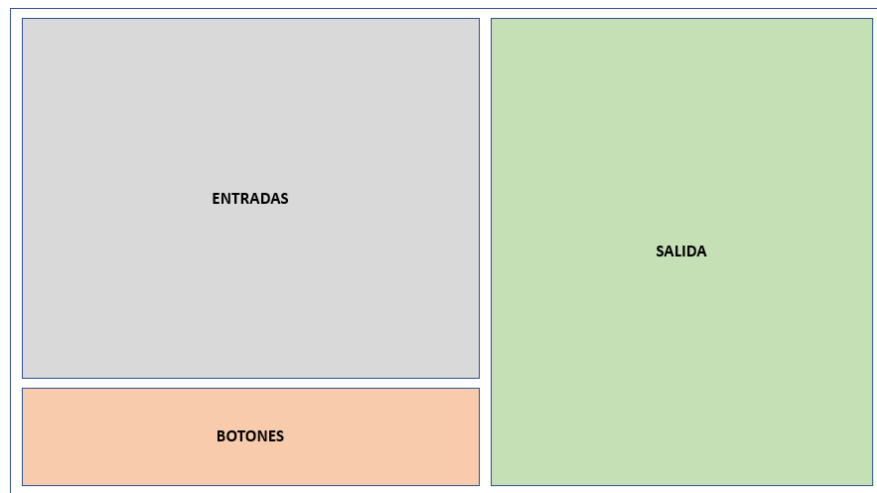


Figura 30: Boceto de la pestaña Principal de la primera versión del software.

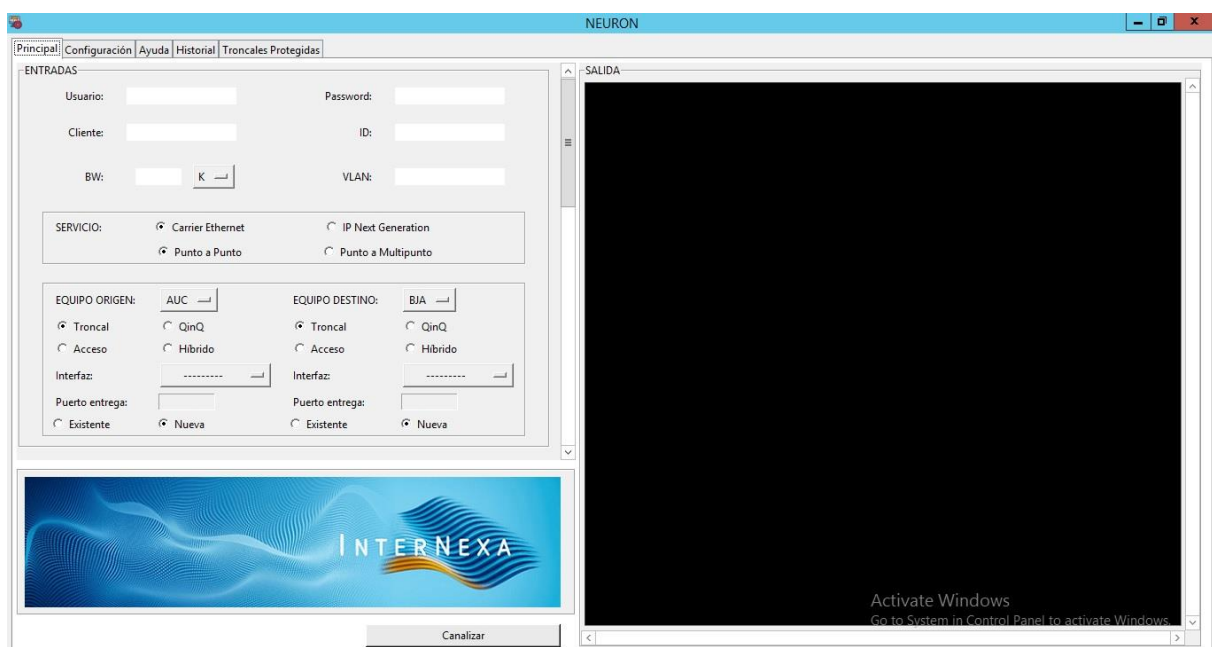


Figura 31: Apariencia real del boceto sugerido por el NOC.

Las entradas corresponden a todos los datos necesarios que necesita un Ingeniero de configuración para la canalización de un servicio ya sea CE o IPNG. Estos datos fueron presentados en las secciones 4.2.1 y 4.2.2. Para el caso en que se desee configurar un servicio IPNG, aparecerán unos campos adicionales los cuales es muestran en la Figura 32.

| | |
|----------|----------------|
| VPRN | Equipo Alcatel |
| Dir IPv4 | LAG/ PUERTO |
| Dir IPv6 | Equipo Destino |

Figura 32: Datos necesarios para un servicio IPNG.

El campo **VPRN** (Virtual Private Route Network) corresponde a un servicio VPN capa 3 usado en los equipos de Alcatel para entregar conexión a múltiples clientes por medio de una única arquitectura de enrutamiento lógico. El campo **Equipo Alcatel** corresponde al router que da acceso a internet. El campo **Equipo Destino** corresponde a un equipo MPLS Huawei que será donde se va a recibir la conexión para el cliente final.

La salida corresponde a toda la información que los equipos a ser configurados entregan al usuario. Después de la ejecución de cada comando, el equipo que está siendo configurado emite una respuesta que es plasmada en ese tablero.

El campo de los botones corresponde al lugar donde se encuentra el botón para iniciar la configuración automática. Durante el proceso de canalización este botón se redibuja para que aparezca otro que permita abortar el proceso.

A parte de la pestaña **Principal**, la Interfaz Gráfica cuenta con otras pestañas que adicionan otras funcionalidades al software. Estas pestañas se mencionan a continuación:

- En la pestaña **Configuración** esta la funcionalidad para agregar o eliminar equipos a la base de datos. Para esto es necesario usar las librerías Threading para crear un hilo aparte durante la ejecución y mysql.connector.
- En la pestaña **Ayuda** se encuentran los manuales de configuración, de usuario y las personas que participaron en la construcción del Software. En esta pestaña se usan las librerías Threading y webbrowser para acceder a documentos PDF.
- En la pestaña **Historial** se encuentran todas las configuraciones realizadas por el programa. En cada fila del listado se encuentran datos como la hora y la fecha de la configuración, la duración, la persona que realizó la configuración y la descripción del servicio. Para acceder a este historial es necesario usar la librería mysql.connector.
- En la pestaña **Troncales Protegidas** se encuentran todos los puertos protegidos en los equipos. Para acceder a esta información en la base de datos, es necesario usar la librería mysql.connector.

La librería usada para desarrollar la GUI es tkinter, la cual proporciona un grupo de herramientas para facilitar la administración de ventanas. Durante mucho tiempo ha sido parte integral de Python. Viene incluida en los sistemas operativos Windows, MacOS y en las distribuciones GNU/Linux [25].

La principal ventaja que ofrece los módulos tkinter es su velocidad. A pesar de que la documentación es deficiente, existen demasiados recursos como bibliografías, tutoriales, libros, entre otros [25].

La primera versión del programa cuenta con los siguientes widgets:

- Botones
- Etiquetas
- Campos de Entradas
- Menú de opciones
- Frames
- Barra de desplazamiento
- Botones Radio
- SpinBox
- Ventanas Emergentes
- Botones de menú
- Cajas con listas
- Pestañas

4.3.1.3. Conexiones seguras

Como se dijo en la sección 3.7 en cuanto a las políticas de TI, las conexiones seguras con los dispositivos se realizan manualmente por SSH. Para las conexiones SSH en Python se utiliza la librería Paramiko, la cual implementa SSH v2 que proporciona funcionalidades para el modelo cliente servidor. Muchos conocen SSH v2 como el protocolo que reemplazo Telnet y rsh para los accesos remotos, pero además cuenta con la capacidad de abrir canales cifrados para servicios como SFTP (Secure File Transfer Protocol) [26].

Viendo las ventajas anteriores que tiene esta librería, representa un gran potencial en el sector de las telecomunicaciones, ya que se pueden automatizar tareas extensas y repetitivas en los dispositivos de red.

A continuación, se describen las tres etapas requeridas para implementar las conexiones remotas por SSH: establecimiento de sesión, envío/recepción de datos y cierre de sesión.

❖ **Establecimiento de Sesión:** para el establecimiento de sesión se debe seguir el siguiente procedimiento:

1. Para crear el cliente SSH se utiliza la instrucción `"conexion=paramiko.SSHClient()"`.
2. Después de crear el cliente, es necesario definir las políticas para la negociación de claves. Para este caso se utiliza la instrucción `"conexion.set_missing_host_key_policy(paramiko.AutoAddPolicy())"`, donde se especifica que no se cuenta con claves públicas y privadas.
3. Luego se realiza la conexión con el servidor con la instrucción `"conexion.connect(ip_gestion, 22, ssh_usuario, ssh_clave)"`. Los parámetros de esta función corresponden a la IP de gestión del

servidor, el puerto donde se va a realizar la conexión, el usuario y la contraseña respectivamente.

4. Finalmente se crea un canal con la instrucción `"canal=conexion.invoke_shell()"`. Utilizando este tipo de canales es posible enviar comandos y recibir salidas del servidor varias veces sin que se cierre la sesión.

❖ **Envío/Recepción de datos:** Para el envío de comandos a los equipos de capa 2-capa 3, en esta primera versión del programa se utiliza la modalidad de conexión basada en una única capa (shell-based) con el fin de que, al crearse la conexión, se creara un canal donde se pudiera enviar comandos y recibir salidas varias veces. En la Figura 33 se puede ver como es el esquema para el envío de comandos basado en única capa.

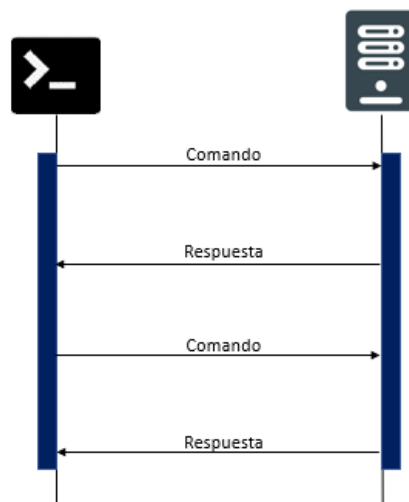


Figura 33: Esquema de funcionamiento shell-based.

Para utilizar este esquema con la librería Paramiko se deben seguir los siguientes pasos:

1. Primero se escribe la instrucción `"channel.send(comando + "\n")"` para enviar el comando por el canal creado. Al agregar el carácter `"\n"` es como si presionáramos la tecla `"Enter"` en un Shell.
2. Después se utiliza la instrucción booleana `"channel.send_ready()"` para garantizar que el comando se haya enviado completamente al servidor.
3. Luego, se utiliza la instrucción booleana `"channel.recv_ready()"`, para verificar si ya el servidor devolvió una respuesta. Retorna falso en el caso de que aún no se tenga respuesta del servidor, entonces el cliente se queda esperando hasta recibir dicha respuesta.
4. Ya cuando se tenga una respuesta se utiliza la instrucción `"output=channel.recv(10192)"` donde ese número corresponde al

número máximo de bytes que se pretenden leer. Ya con la respuesta almacenada en una variable se hace la respectiva decodificación y procesamiento.

Todos esos pasos se repiten con cada uno de los comandos hasta que no haya más.

- ❖ **Cierre de Sesión:** Después de que se hayan enviado todos los comandos necesarios durante la canalización, es necesario cerrar las conexiones con cada uno de los equipos. Para esto hay que utilizar el método “.close()” de la librería Paramiko. Por ejemplo, si se configuraron dos equipos durante la canalización, las instrucciones a utilizar son “conexion1.close()” y “conexion2.close()”.

4.3.2. Aplicación de los cambios en el software

Luego de haber estudiado el funcionamiento del software se llegó al acuerdo de que no se iba a crear un software desde cero ya que la primera versión estaba bien estructurada y el personal de la empresa ya estaba familiarizada con su uso. Entonces lo que se tenía que hacer era aplicarle los cambios para que se adaptara a la nueva situación de la compañía.

En esta etapa se establecieron encuentros para evaluar periódicamente el avance del desarrollo del software.

A continuación, en la Tabla 1 se listan los ajustes realizados:

Tabla 1: Listado de modificaciones realizadas por el software.

| Características | Cambios Aplicados |
|-----------------------------|---|
| GUI | Se agregaron Ventanas Emergentes. |
| | Se mejoraron las salidas mostradas en el tablero de resultados. |
| | Se eliminaron de la GUI los elementos relacionados con configuración de puertos híbridos. |
| | Cambio de la presentación de la información mostrada en el listado de equipos. |
| | Se agregaron widgets que facilitan el trabajo de los usuarios. |
| | Se agregaron etiquetas de ejemplo que hacen que el software sea más fácil de usar |
| | Todos los campos al finalizar una configuración quedan vacíos. |
| | Actualización del listado de equipos en tiempo de ejecución en la pestaña de Configuración. |
| | Organización del historial de configuraciones realizadas de manera cronológica en la pestaña de Historial. |
| | Creación de un filtro que permita buscar una o varias configuraciones por palabra clave en la pestaña de Historial. |
| Implementación de Servicios | Cambio en las validaciones iniciales. |
| | Entrega de servicios por medio de subinterfaces. |

| | |
|---------------------|---|
| | Nueva descripción en la interfaz y subinterfaz. |
| | Cambios en la prueba del servicio en el caso de VSI. |
| | Manejo de errores en el caso de comandos fallidos. |
| | Corrección en cuanto a la búsqueda del túnel MPLS. |
| | Cambio en la configuración de servicios L2 VPN. |
| Escritura de código | Utilización de la convención PEP8. |
| | Eliminación de código relacionada con configuraciones híbridas. |
| | Creación de nuevas funciones para tareas repetitivas. |
| | Actualización en la documentación de las funciones. |

4.3.2.1. GUI

- Se adicionaron ventanas emergentes de advertencia para informar al usuario de eventos como: número de VLAN ocupada en un determinado equipo, configuración VSI incompleta y configuración VLL incompleta.
- Se mejoró el formato de las salidas mostradas para mejorar la experiencia de usuario. Por ejemplo, anteriormente el software al momento de conectarse remotamente con un equipo solo se mostraba en el tablero de resultados la IP de gestión. Esta información a menudo era muy limitada para el usuario, por tal motivo se agregó un campo que indica a cuál equipo se está realizando la conexión remota. Otro caso es que solo mostraba mensajes para la validación de disponibilidad VLAN, ahora lo hace tanto para VLAN como de subinterfaz.
- Los Ingenieros de Configuración explicaron que los servicios con configuraciones de interfaz híbrida son demasiado raros, por esa razón se eliminó los radiobutton referentes a esa opción. Esta modificación se puede ver en la Figura 34 y en la Figura 36.
- En cuanto al cambio de la presentación de la información de los equipos en el listado, anteriormente se mostraba el nombre abreviado del equipo como se puede ver en la Figura 31, pero debido al aumento de equipos en la red últimamente ocurrían confusiones. Por esa razón, en esta versión los nombres mostrados en el listado corresponden al nombre completo del equipo. Esta modificación se puede ver en la Figura 34, Figura 35, y en la Figura 36
- Un widget que se agregó es una barra de desplazamiento horizontal debido a que el tamaño de los frame aumentó. También se agregó contorno negro a las entradas de texto porque anteriormente se confundían con el fondo del programa (Ver Figura 34, Figura 35, Figura 36 y Figura 37). La utilidad de la barra de desplazamiento horizontal se puede ver en la Figura 36 para cuando se va a configurar un VSI.
- Se agregaron etiquetas explicativas al software porque en el caso de configurar un servicio IPNG, los usuarios no sabían de qué manera ingresar la dirección IPv4 con su máscara. Entonces debajo de ese campo de entrada de texto se agregó una etiqueta donde mostraba que el formato debía ser de la siguiente manera: xx.xx.yy.yy/zz. Se hizo también una etiqueta explicativa

para el formato de la dirección IPv6. Estas etiquetas se pueden ver en la Figura 35.

- Por recomendación de uno de los Ingenieros de Configuración, se hizo posible que los campos de ingreso de texto se vaciaran, para evitar errores en el caso de que el usuario quisiera hacer otra configuración inmediatamente.
- En la pestaña de Configuración de la primera versión (ver Figura 37), está la funcionalidad de agregar o eliminar equipos de la Base de Datos, pero para ver reflejada esta modificación en el listado de equipos tocaba cerrar y volver a ejecutar el programa. Para ahorrarle tiempo al usuario se agregó la posibilidad de que los cambios realizados en la base de datos se vieran reflejados en tiempo de ejecución.
- El historial que se encontraba en la primera versión del software mostraba las configuraciones realizadas de la más antigua a la más reciente, algo que no es usual en los aplicativos actuales. Como no fue posible realizar modificaciones en la Base de Datos por políticas de la empresa, entonces lo que se hizo fue invertir esa lista en Python.
- En la pestaña del historial se agregó una funcionalidad que permitía buscar configuraciones por palabra clave ya que el listado cada vez se hace más extenso. En la Figura 38 se puede ver el resultado de dicha adición.

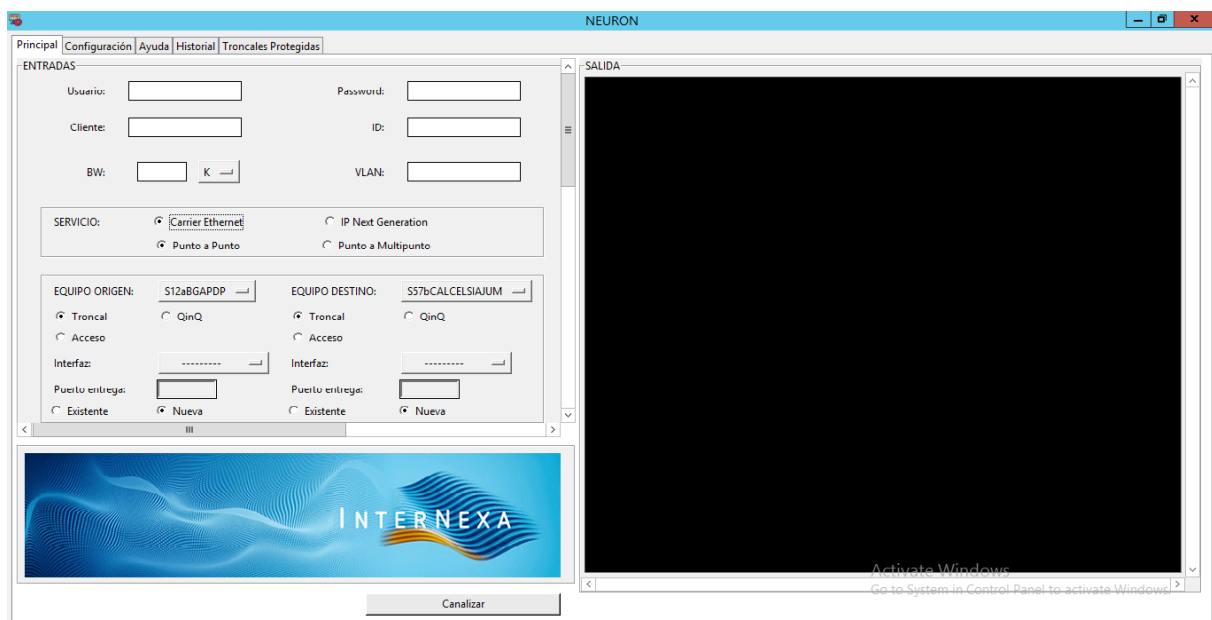


Figura 34: GUI principal con modificaciones.

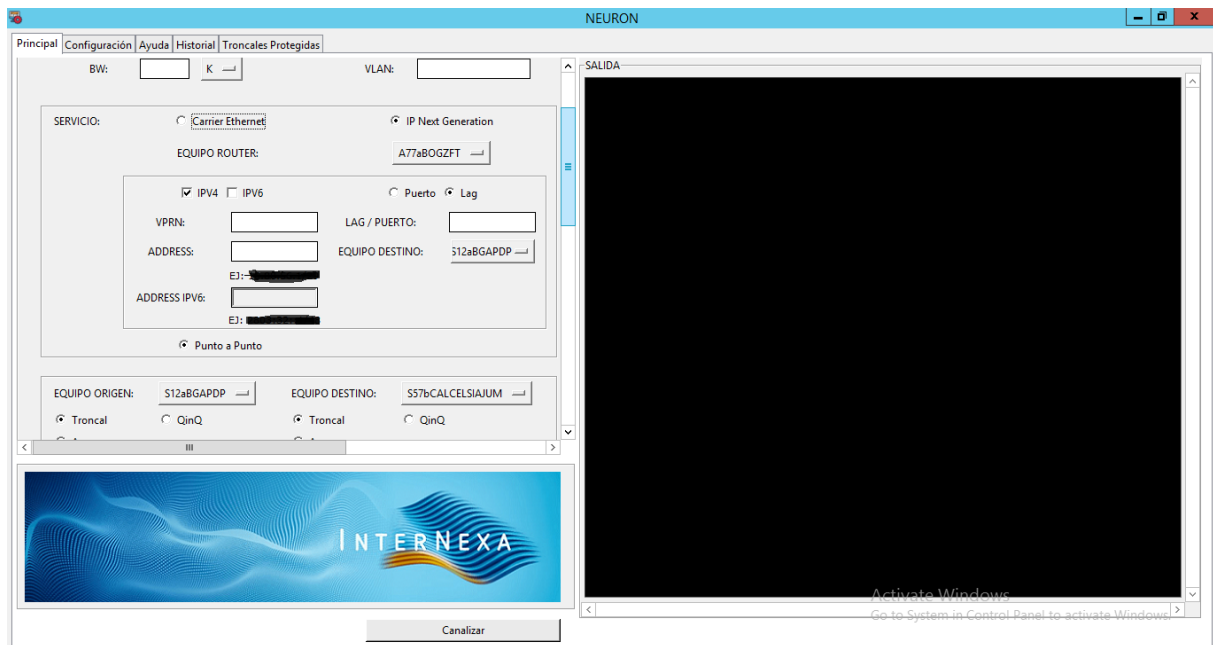


Figura 35: Cambios en el menú de IPNG.

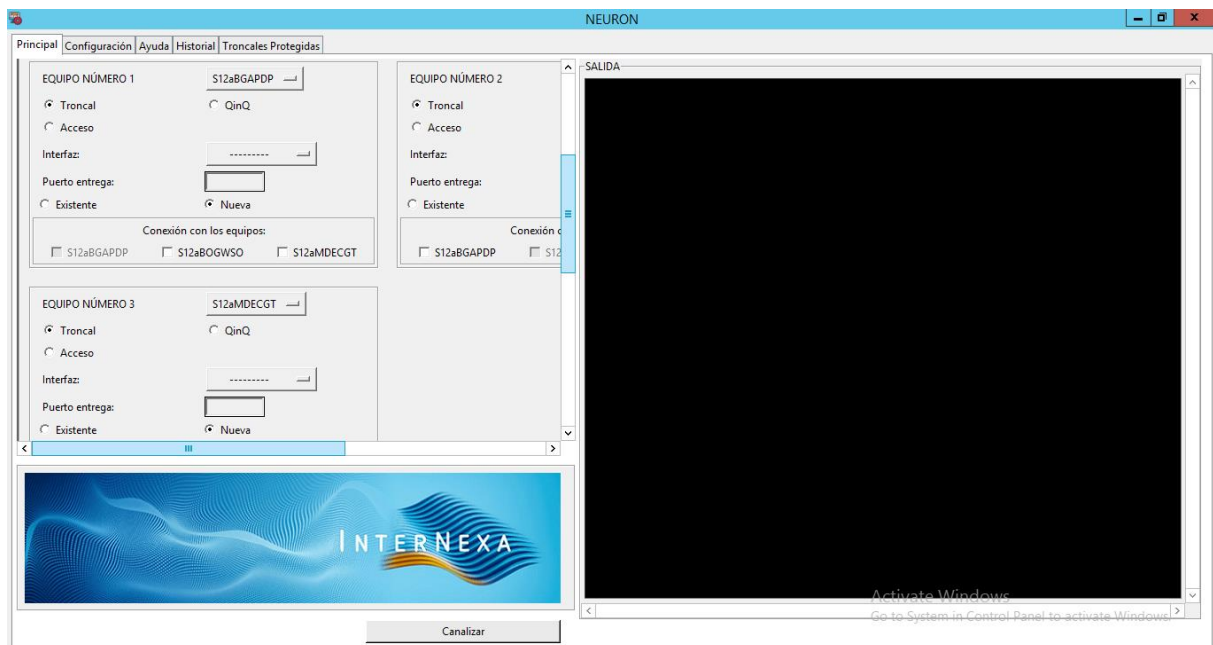


Figura 36: Cambios en la GUI en VSI.

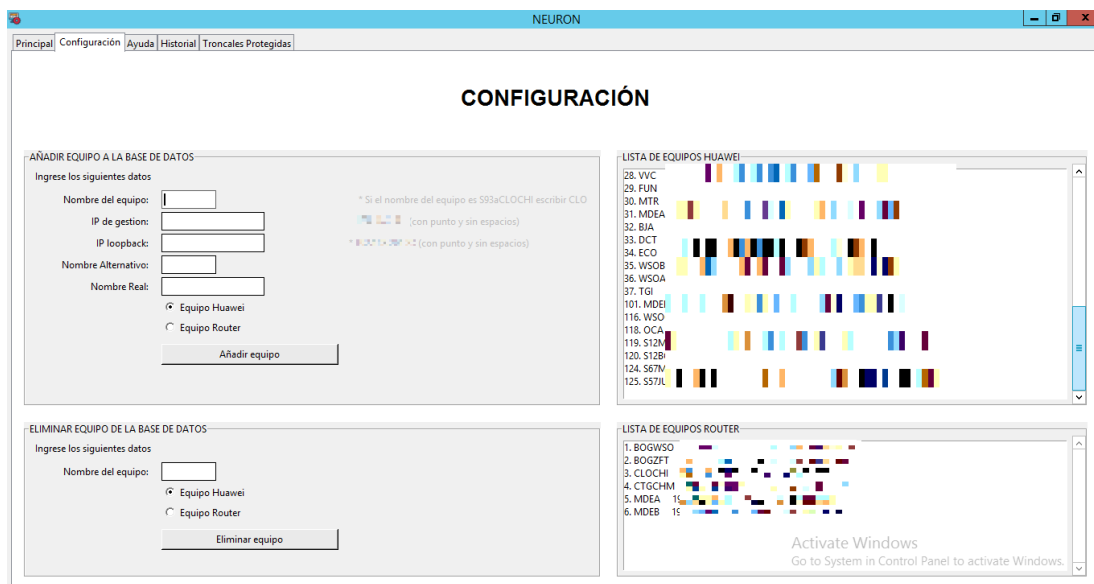


Figura 37: Ventana de configuración modificada.

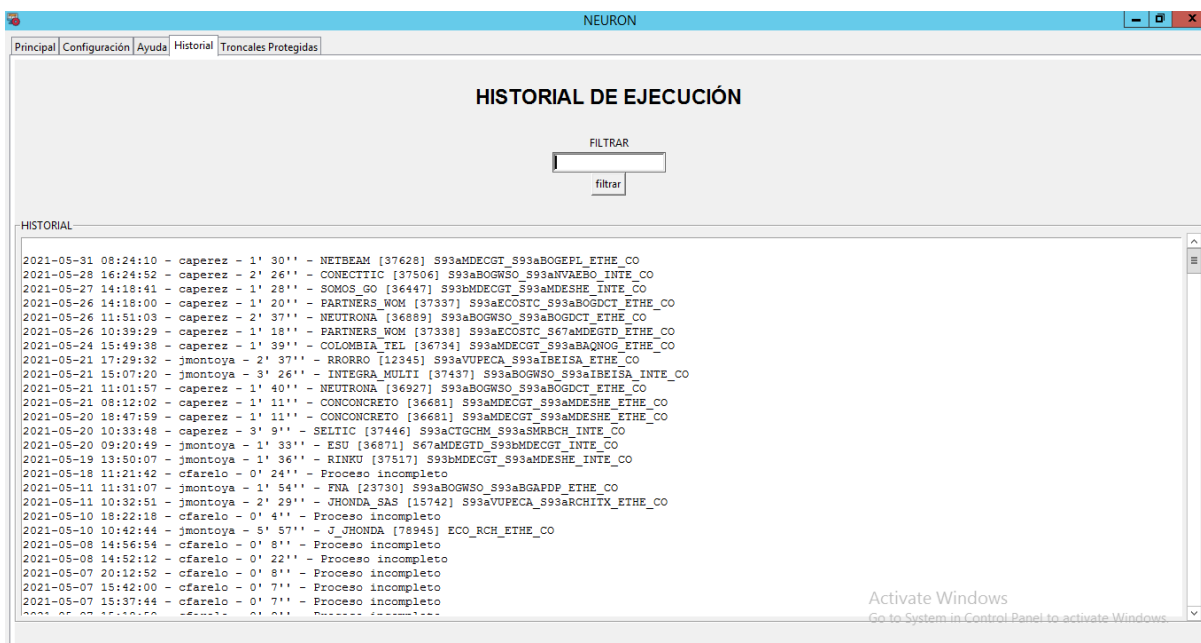


Figura 38: Ventana de Historial modificada.

4.3.2.2. Implementación de Servicios

- Anteriormente, el software hacía validación del formato del puerto, de la dirección IP y de la disponibilidad de la VLAN. Ahora como los servicios se entregan por medio de subinterfaces, es necesario que además de las validaciones anteriores, también valide la disponibilidad de la subinterfaz.
- Anteriormente los puertos y las interfaces VLAN se les agregaba una descripción donde llevaba las palabras CE o IPNG dependiendo del servicio. Ahora la descripción va es en el puerto y en la subinterfaz. Además, los servicios

CE ahora se describirán como ETHE_CO mientras que los servicios IPNG se describen como INTE_CO.

- Se implementaron nuevas funciones en el código para habilitar la configuración de servicios CE e IPNG por medio de subinterfaces, para cumplir con los requerimientos de TI de la compañía.
- La modificación que se hizo en la parte de la prueba del servicio en configuraciones VSI fue agregar dos comandos de prueba que mostraban información más sencilla. También se hacían recorridos repetitivos en los equipos que tomaban más tiempo, pero garantizaban que el servicio quedara bien establecido.
- En algunos equipos estaba ocurriendo un error de interpretación de comandos y por cuestiones relacionadas con el presupuesto no era posible cambiar el equipo. Para solucionar esto utilizo un ciclo while para volver a enviar el comando hasta que el mensaje de error desapareciera. En los casos donde se presentaban estas fallas era con los errores de TACACS y con comandos que llegaban incompletos.
- Una de las fallas que se descubrió en la primera versión del software era que asignaba un túnel MPLS incorrecto cuando veía que había una IP de loopback asociada a este túnel parecida a la correcta. Entonces para corregirlo se realizó una comparación de exactitud entre la IP loopback asociada al túnel y la obtenida desde la base de datos.
- Para esta nueva versión del software, el Especialista de Red de la compañía le había enseñado a los Ingenieros de Configuración como configurar servicios L2 VPN en puertos QinQ utilizando subinterfaces. Entonces un día se realizó una prueba con un servicio ficticio con dos equipos ubicados en la Costa utilizando dicha configuración, pero lastimosamente se presentó un loop en los equipos durante la prueba del servicio por lo cual este caso se llevó a revisión. Luego de más de un mes de estudios del Especialista de Red con Huawei se llegó a la conclusión de una nueva forma de configurar estos servicios sin que se presenten loops y estos cambios fueron aplicados en el código dando como resultado el día de las pruebas una configuración exitosa.

4.3.2.3. Escritura de Código

- Al ver el código de la primera versión se notaba que era un poco desordenado lo cual causaba que fuera difícil de entender para una persona nueva. Entonces después de buscar exhaustivamente sobre buenas prácticas de programación en Python se encontró la convención PEP8, que no afecta el funcionamiento del programa, pero si ayuda a que el código sea mucho más legible. La convención PEP8 da recomendaciones en aspectos como la longitud máxima de una línea, la indentación en los argumentos de una función, los espacios en blanco, entre otros aspectos. En la Figura 39 se puede ver, con ayuda de la herramienta online pep8online.com, que esta versión del software cumple con los requisitos de esta convención.

All right

Save ▾ Share

Your code

```
1 def validacion_ip_add(Flag_Abortar_Todo, Address):
2     if Address != "":
3         if Address.count(".") == 3 and Address.count("/") == 1:
4             Octeto = Address.split(".")
5             UltimoOcteto = Octeto[3].split("/")
6             if (int(Octeto[0]) > 0 and int(Octeto[0]) < 255 and
7                 int(Octeto[1]) > 0 and int(Octeto[1]) < 255 and
8                 int(Octeto[2]) > 0 and int(Octeto[2]) < 255 and
9                 int(UltimoOcteto[0]) > 0 and int(UltimoOcteto[0]) < 255 and
10                int(UltimoOcteto[1]) > 0 and int(UltimoOcteto[1]) < 32):
11                 print("Formato ADDRESS correcto")
12                 Flag_Abortar_Todo = False
13             else:
14                 Flag_Abortar_Todo = True
15                 messagebox.showinfo(message="¡EL FORMATO DE DIRECCIONAMIENTO "
```

Check again

Figura 39: Verificación de la convención PEP8 del código.

- Como los Ingenieros de configuración habían mencionado que las configuraciones híbridas eran poco frecuentes, entonces se eliminaron las partes del código relacionadas con ese aspecto lo cual representaba alrededor de 500 líneas de código. En el caso de Python que es un lenguaje interpretado, eliminar toda esa cantidad de líneas innecesarias es beneficioso.
- Validar el formato de los puertos y de la dirección IP es una tarea muy frecuente en el software, entonces para aplicar buenas prácticas de programación fue necesario crear una única función que solo se llamara cuando fuera necesario.
- Inicialmente la documentación de cada función se hacía aplicando el estándar PEP257 (ver ejemplo en la Figura 40), lo cual es muy útil a la hora de leer inicialmente el programa. Como las funciones iban recibiendo nuevos argumentos, realizaban nuevas tareas y retornaban otras variables, fue necesario actualizar esta documentación.

```
def multiply(a, b):
    """This method multiplies the given two numbers.

    Input Arguments: a, b must be numbers.
    Returns: Multiplication of a and b.
    """
    return a * b
```

Figura 40: Ejemplo de la convención PEP257.

Tomado de <https://www.askpython.com/python/python-docstring>

4.3.3. Descripción del Software Desarrollado

En este apartado se procederá a realizar una descripción del software como tal. Como este proyecto es muy complejo no basta con capturar su arquitectura en un solo diagrama, es por lo cual se va a emplear el Modelo de Vistas de Kruchten 4+1 explicado en la sección 3.8.2. Este modelo representa todas las características de la arquitectura en múltiples vistas y por esta razón es el más utilizado en el desarrollo de software.

Para este proyecto, los componentes mostrados en la Figura 21 son los siguientes:

- El **sistema** en el que se trabajo está compuesto por el software de configuración automática para servicios CE e IPNG, los switches de enrutamiento Huawei, los enrutadores Alcatel, las redes de Fibra Óptica, el servidor TACACS de la compañía y la Base de Datos.
- Este sistema cuenta con varios **Interesados** los cuales son los directivos de Internexa, los Ingenieros del NOC y el estudiante en práctica.
- Cada uno de los interesados tiene sus **Preocupaciones**, los directivos de la compañía están preocupados por los aspectos económicos y de macroprocesos. Los Ingenieros del NOC están preocupados por el tiempo que demora la configuración automática y por la experiencia de usuario que brinda el software. El estudiante en práctica está preocupado por el funcionamiento del software para su proceso de graduación.
- La **Arquitectura** del sistema corresponde a todos los elementos y el relacionamiento existente entre ellos.
- Por último, se encuentra la **Descripción de la Arquitectura** que está compuesta por los **Puntos de Vista** que explican las preocupaciones de los interesados y la **Vista** correspondiente.

La descripción del software en cada una de las vistas se muestra a continuación:

4.3.3.1. Vista Lógica

Con esta vista se representa el funcionamiento general del programa. En la Figura 41 se puede ver que el usuario tiene la capacidad de conectarse por medio de SSH a los equipos Huawei y Alcatel, enviar comandos y recibir resultados por parte de los equipos. Por su parte los dispositivos de enrutamiento envían los datos generados por el usuario al servidor TACACS para realizar la validación correspondiente y recibir esa respuesta.

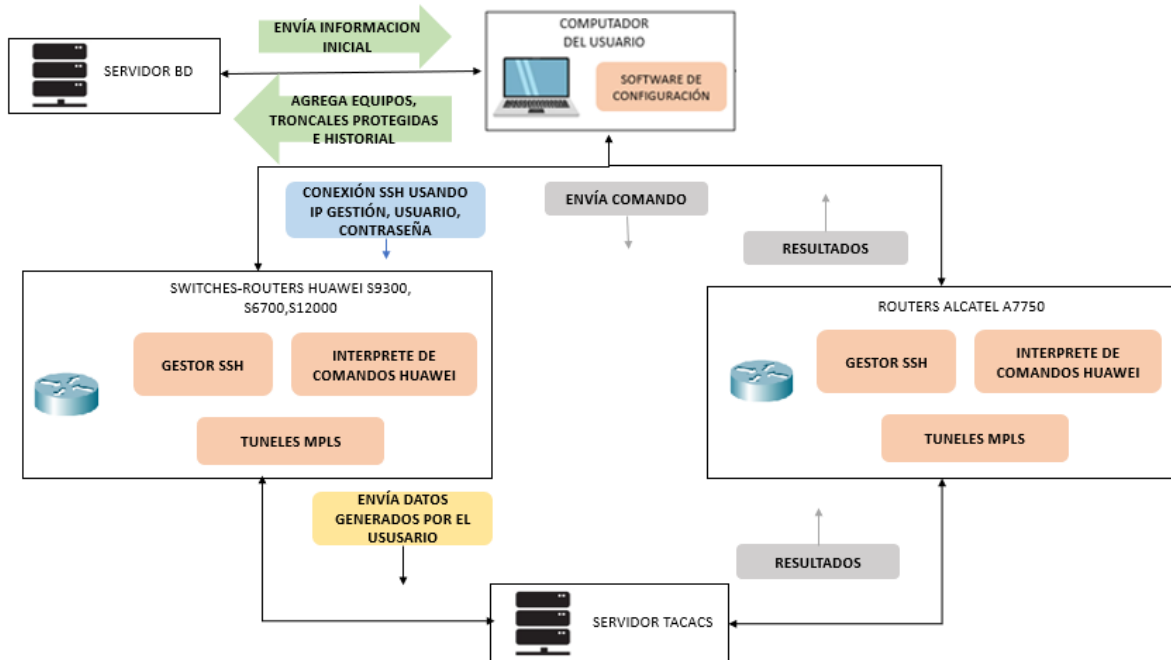


Figura 41: Vista Lógica del software.

4.3.3.2. Vista de Despliegue

En esta vista se muestra las componentes en las que está dividido el software y las dependencias que tiene cada una. En la Figura 42 se puede ver los paquetes y algunos de los métodos usados en el software de configuración automática.

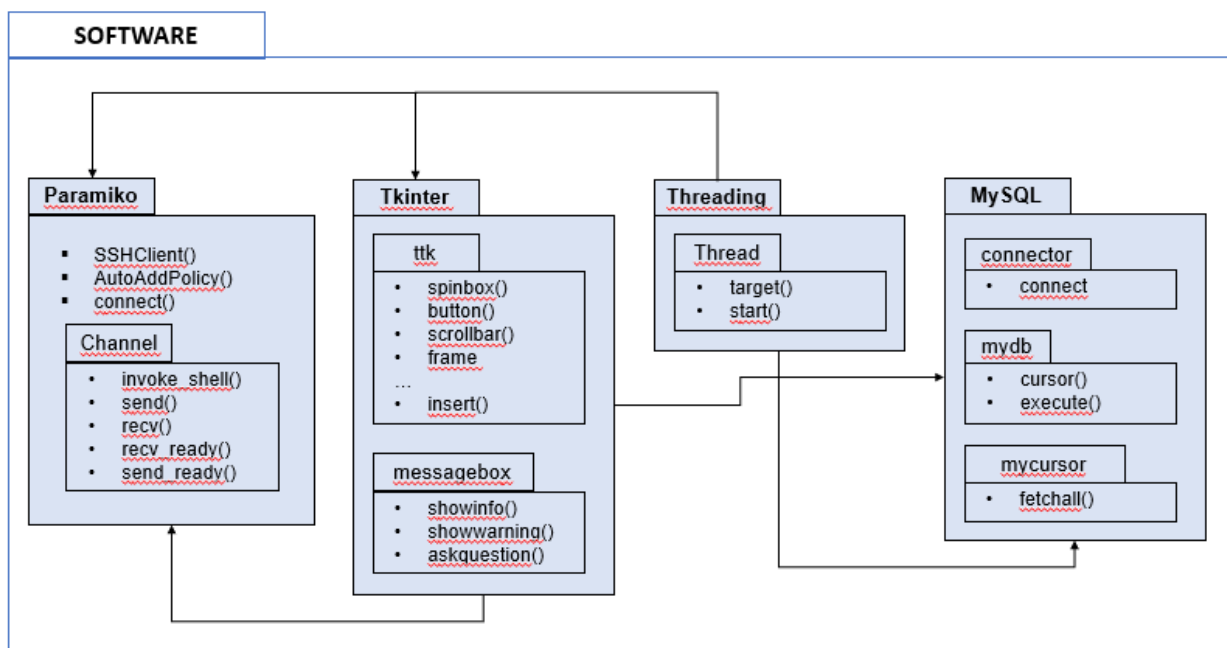


Figura 42: Vista de Despliegue del Software.

4.3.3.3. Vista de Procesos

En esta vista se realiza la descripción de la arquitectura desde la perspectiva de un integrador de sistemas. En la Figura 43 se puede ver la secuencia de procesos resumida y los diferentes elementos del sistema realizando cada uno de esos procesos. En la Figura 43 también se puede ver que la mayoría de las tareas las realiza el software de configuración automática, lo cual era lo que se buscaba durante el proyecto, pero si no hay iniciación por parte del usuario ni la disponibilidad de los equipos en la red, estas configuraciones no serían posibles.

En el diagrama también se puede ver que el software realiza acciones ante posibles fallas que puedan ocurrir durante la ejecución del programa.

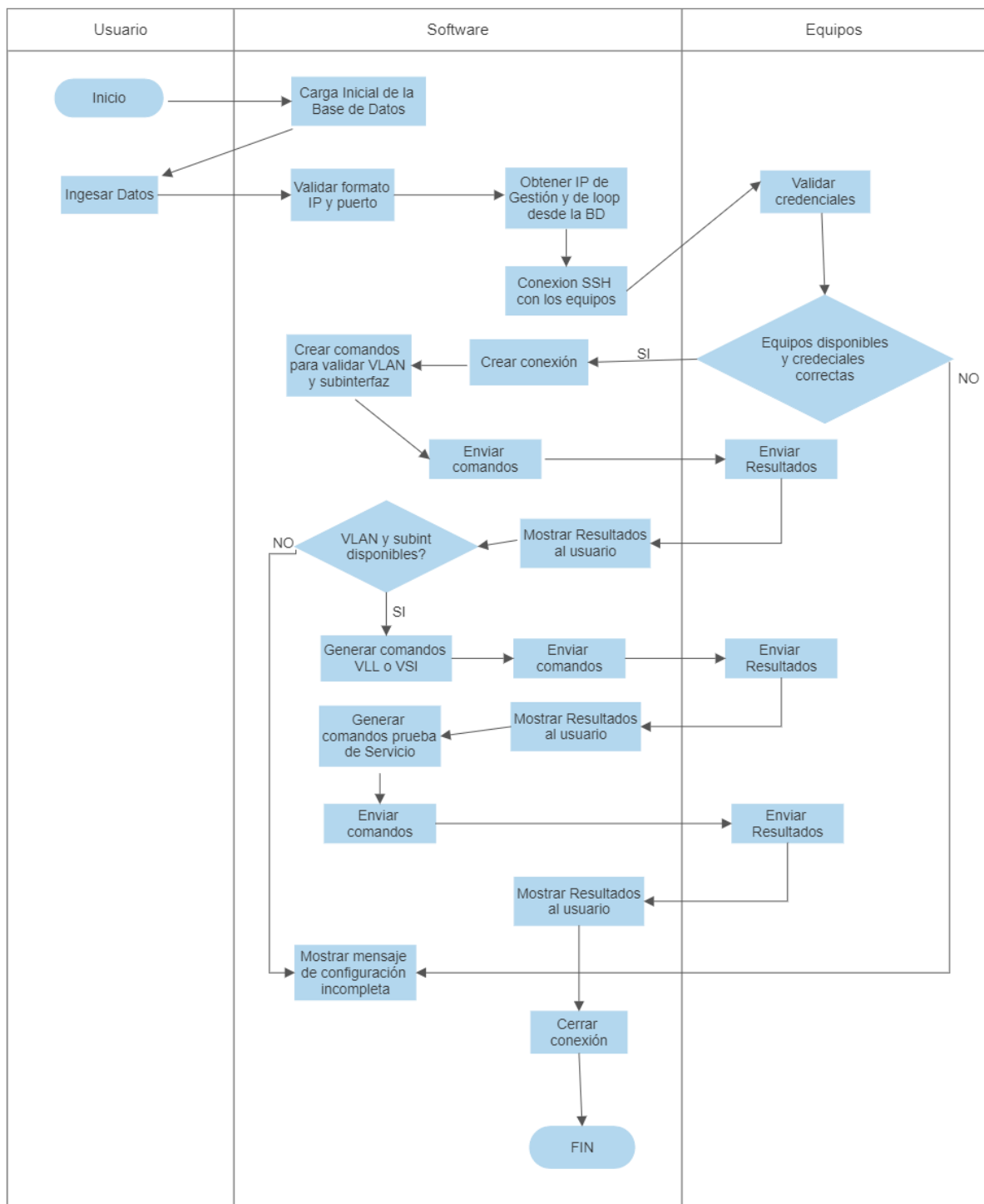


Figura 43: Diagrama de actividades del software.

4.3.3.4. Vista Física

En esta vista se hace la descripción de la arquitectura desde la perspectiva de un Ingeniero de Red. Acá se presentan todos los componentes físicos involucrados y su forma de comunicarse. En la Figura 44 se puede ver todos los componentes físicos que estuvieron involucrados durante el desarrollo del software, la descripción general de cada elemento y la forma de comunicarse entre ellos.

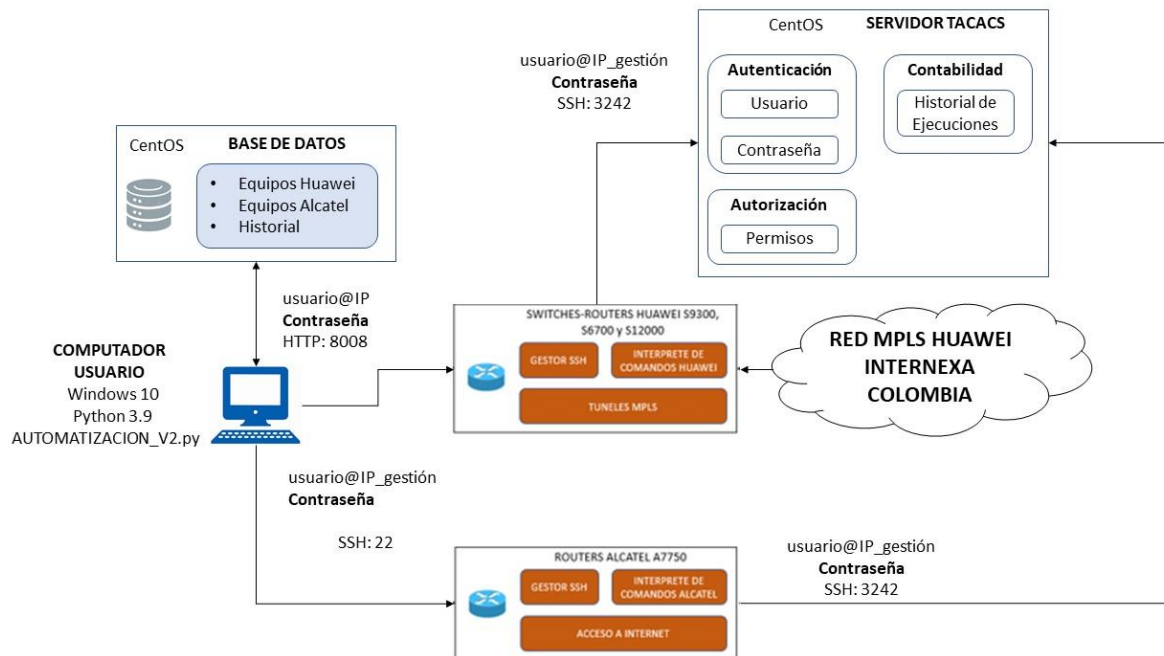


Figura 44: Vista Física de la arquitectura.

4.3.3.5. +1 Vista de Escenarios

En esta vista se proporcionan las funcionalidades desde todos los puntos de vista involucrados. En la Figura 45 se muestra el diagrama de casos para el software desarrollado.

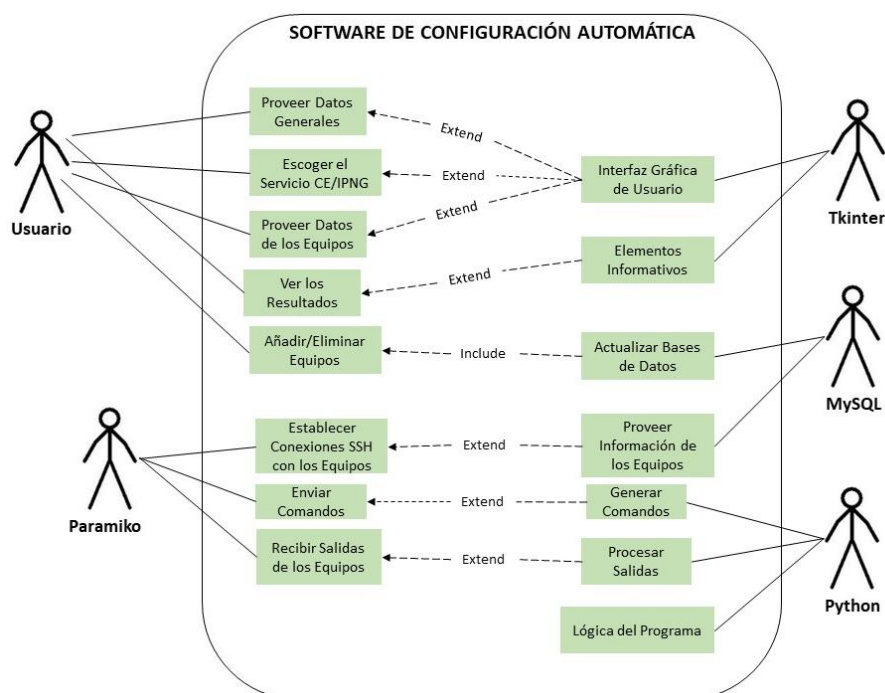


Figura 45: Vista de Escenarios del Software.

4.4 Aprobación y entrega del software

Esta etapa del proyecto consistió en una etapa del desarrollo correspondiente a las pruebas de usuario, actualización de los manuales, toma de tiempos de configuración y entrega oficial del software a la compañía.

4.4.1. Pruebas de usuario

Ya después de haber realizado pruebas exitosas de servicios ficticios en equipos alejados de la red MPLS con diferentes tipos de enlace, se procedió a entregar el software a los Ingenieros de Configuración para que pudieran optimizar sus labores dentro de la compañía.

Como la mayoría del personal de la compañía no cuenta con IDE (Entorno de Desarrollo Integrado) de Python como por ejemplo Spyder o Visual Studio Code, fue necesario crear un ejecutable que contenga todas las librerías que requiere el programa. Para crear este instalador, se utilizó un paquete de Python llamado **pyinstaller**, el cual tiene la ventaja de ser fácil de usar y compatible con el sistema operativo Windows.

Durante esta fase del proyecto, los configuradores, a medida que usaban el software descubrieron que había pequeños errores que eran necesario ajustar, como por ejemplo la descripción en los equipos Alcatel y la funcionalidad del filtro para que no solo buscara un servicio por ID sino por cualquier palabra clave. Durante cualquier proyecto de Desarrollo de Software es importante esta etapa de pruebas de usuario antes de realizar la entrega oficial, porque el programa está pensado para que lo usen otras personas.

4.4.2. Realización de los manuales

Ya después de haber modificado el programa de acuerdo con los nuevos requerimientos de la compañía, es necesario documentar los instructivos para los usuarios. La primera versión contenía el manual de usuario y de configuración, sobre los cuales se realizaron las modificaciones.

4.4.2.1. Modificación del manual de configuración

Este manual es pensado para el desarrollador, ya que contiene la explicación de la convención PEP 257, la documentación de las funciones y el acceso a las bases de datos. A continuación, se muestra los cambios realizados a este manual:

- Se actualizó la documentación de las funciones usadas ya que en todas variaron los argumentos y las variables de retorno. Además, también hubo necesidad de definir nuevas funciones.
- Se agregó información acerca de la convención PEP8, la cual es usada en el código para hacerlo más legible.
- Se tomaron pantallazos de todas las pestañas del software para indicar el nombre de las variables de los widgets. Esta parte es muy importante para facilitar el trabajo de la próxima persona que vaya a trabajar en el proyecto, ya que se tiene pensado que este software es escalable para otras operaciones dentro de la compañía.

- Se indicó el paso a paso para crear el archivo ejecutable, lo cual ayudará también a los usuarios en el caso de que este tipo de archivos sea borrado por políticas de seguridad de la empresa.

4.4.2.2. Modificación del manual de usuario

Este manual de usuario es pensado para los Ingenieros de Configuración y de Gestión quienes son los que sacarán el máximo provecho al software de configuración automática. A continuación, se muestran las modificaciones realizadas:

- En las secciones del software se agregaron los nuevos pantallazos del actual diseño de la interfaz gráfica.
- En cuanto a la parte de la configuración de los diferentes tipos de servicio, se indicaron los nuevos pasos para su uso. Además, se agregaron capturas de las salidas resultantes en cada caso, los cuales son de gran ayuda para que el usuario tenga conocimiento de lo que está pasando en los equipos durante la ejecución.
- Se explica la manera de añadir o eliminar un equipo, basado en la nueva nomenclatura y en la actualización del listado de equipos en tiempo de ejecución.
- Se agregaron capturas de los nuevos mensajes de advertencia con los que cuenta este software.

4.4.3. Toma de tiempos

Una de las formas que hay para medir el impacto de este software dentro de la compañía, es la toma de los tiempos para cada tipo de servicio. Gracias a que el software cuenta con una pestaña de historial, diariamente se revisaba que servicios reales eran configurados.

Los Ingenieros de Configuración entregaron los datos de cuánto tiempo les tomaba realizar su labor de manera manual, basados en sus años de experiencia.

Con ambos tiempos obtenidos, se realizó una tabla comparativa y un gráfico de líneas en el cual se evidencia la diferencia abismal de tiempos.

4.4.4. Entrega oficial del software

Después de haber culminado las etapas mencionadas anteriormente, ya los Ingenieros de Configuración dijeron que el software estaba listo para ser entregado de manera oficial a la compañía. Para esto se hicieron reuniones donde se convocaron a diferentes áreas del NOC donde primero se hacía una explicación de todo lo que se puede hacer con el programa, luego se respondían las dudas que se tuvieran al respecto y finalmente se hacían configuraciones de servicios ficticios para que todos los presentes vieran como era el funcionamiento del programa.

Además, se mostraron tablas y gráficos comparativos donde es muy notoria la reducción de los tiempos en comparación con la configuración manual.

Los especialistas de red al finalizar la reunión dieron su aprobación y sugerencias acerca de nuevas funcionalidades que pueden ser agregadas en una próxima versión.

5. Resultados y análisis

Para medir el impacto del software realizado dentro de la compañía, se tuvieron en cuenta aspectos como los tiempos de configuración, experiencia de usuario, escalabilidad y mejoras en los servicios prestados.

5.1 Tiempos de Configuración

La principal ventaja que ofrece este software se refleja en el tiempo empleado para configurar un servicio CE o IPNG, el cual es mucho menor comparado con la configuración realizada de manera manual.

Para visibilizar este aspecto se tomaron del historial la duración que tuvo cada una de las configuraciones en diferentes equipos y los tiempos de configuración manual en cada una de las variantes de servicio. Sin embargo, el tiempo obtenido a través del historial del software solo toma en cuenta el tiempo de ejecución del programa, por lo que fue necesario ajustar a este tiempo, el tiempo que tardan los Configuradores en diligenciar los campos requeridos para la configuración de un servicio.

Para la comparación de tiempos, solo se mostrarán los casos de CE punto a punto y de IPNG, debido a que estos dos casos han sido los más demandados por los clientes durante estos últimos meses. También es importante mencionar que, una configuración CE punto a punto es más rápida que un IPNG debido a que en el primer caso solo se hace conexión con dos equipos MPLS Huawei, mientras que en IPNG es necesario conectarse a los 2 equipos MPLS de Huawei y el router de Alcatel.

5.1.1. Tiempo de establecimiento de un servicio CE VLL

Para este caso, se tomaron 8 tiempos de configuración del software para clientes reales en diferentes localidades del país. Los tiempos manuales en cada caso fueron brindados por los Ingenieros de Configuración basados en su experiencia.

Para llenar los campos requeridos en un CE VLL los configuradores se tomaron aproximadamente entre 1:50 y 2:20 minutos. El tiempo total de la configuración automática dependerá del parámetro anterior, la ubicación del equipo y su estado.

Cabe resaltar que en los tiempos recolectados existen diferentes casos de tipo de enlace y configuraciones. En la Tabla 2 se puede ver los tiempos tomados para cada caso.

Tabla 2: Resultados de los servicios CE VLL.

| SERVICIO CE - PUNTO A PUNTO | | | |
|-----------------------------|--------------------------|--------------------------|---------------------------------------|
| Config Servicio | CONFIGURACION MANUAL | | Tiempo Configuración Automática (min) |
| | Tiempo Ingeniero A (min) | Tiempo Ingeniero B (min) | |
| Servicio 1 | 22 | 23 | 4,62 |
| Servicio 2 | 23 | 20 | 3,83 |
| Servicio 3 | 24 | 22 | 3,18 |

| | | | |
|----------------------------|-------------|--------------|-------------|
| Servicio 4 | 20 | 20 | 4,25 |
| Servicio 5 | 21 | 23 | 4,67 |
| Servicio 6 | 20 | 22 | 5 |
| Servicio 7 | 24 | 21 | 3,5 |
| Servicio 8 | 22 | 20 | 3,33 |
| PROMEDIO | 22 | 21,38 | 4,05 |
| DESVIACIÓN ESTANDAR | 1,60 | 1,30 | 0,68 |

Para sacar una relación porcentual del tiempo optimizado por el software es necesario emplear la Ecuación 1 y la Ecuación 2.

$$TPromManual = \frac{TPromA + TPromB}{2}$$

Ecuación 1: Tiempo promedio de la configuración manual.

$$VlrPorcentual = \frac{TPromAuto}{TPromManual} * 100$$

Ecuación 2: Porcentaje de relación entre la configuración manual y automática.

Para este caso los tiempos de configuración son los siguientes:

TPromA = 22 min

TPromB = 21.38 min

TPromManual = 21.69 min

TPromAuto = 4.05 min

VlrPorcentual = 18.66%

De acuerdo con el valor porcentual calculado para este caso, se está reduciendo el tiempo de configuración en un 81.34%.

5.1.2. Tiempos de establecimiento para un servicio IPNG

Para este caso también se tomaron 8 muestras de servicios de clientes reales gracias a la sección de historial del software. Los tiempos manuales fueron dados también por los Ingenieros de Configuración.

Para llenar los campos requeridos en la interfaz gráfica, los usuarios demoraron entre 2:30 a 2:50 minutos, debido a que aparecen otros campos adicionales los cuales fueron explicados en la sección 4.3.1.2. Al igual que los tiempos tomados en el caso CE Punto a Punto, esos minutos y segundos se sumarán al obtenido por el software para obtener el tiempo total de configuración automática.

Cabe resaltar que en los tiempos recolectados existen diferentes casos de tipo de enlace y configuraciones.

Tabla 3: Tiempos de Configuración IPNG.

| SERVICIO IPNG - PUNTO A PUNTO | | | |
|-------------------------------|--------------------------|--------------------------|---------------------------------------|
| Config Servicio | CONFIGURACION MANUAL | | Tiempo Configuración Automática (min) |
| | Tiempo Ingeniero A (min) | Tiempo Ingeniero B (min) | |
| Servicio 1 | 30 | 29 | 5 |
| Servicio 2 | 28 | 28 | 4,1 |
| Servicio 3 | 25 | 26 | 4,05 |
| Servicio 4 | 26 | 28 | 5,65 |
| Servicio 5 | 31 | 27 | 5,93 |
| Servicio 6 | 29 | 27 | 3,97 |
| Servicio 7 | 31 | 25 | 4,93 |
| Servicio 8 | 31 | 28 | 4,77 |
| PROMEDIO | 29 | 27,25 | 4,80 |
| DESVIACIÓN ESTANDAR | 2,36 | 1,28 | 0,74 |

Para este caso también se utiliza la Ecuación 1 y la Ecuación 2. Los tiempos son los siguientes:

$TPromA = 29 \text{ min}$
 $TPromB = 27.5 \text{ min}$
 $TPromManual = 28.06 \text{ min}$
 $TPromAuto = 4.8 \text{ min}$
 $VlrPorcentual = 17.10 \%$

Para este caso de acuerdo con el Valor Porcentual, se logró una reducción de tiempos del 82.90%.

En los procedimientos de configuración manual, los valores de la desviación estándar son mayores con respecto a la configuración automática, ya que en el primer caso la tendencia a cometer errores es mayor y es más común encontrar valores atípicos.

Finalmente, después de haber estudiado todos esos casos, se presenta un resultado global mostrado en la Tabla 4, donde es posible observar los valores porcentuales de la configuración automática con respecto al tiempo de configuración manual.

Tabla 4: Porcentajes obtenidos con respecto al tiempo manual.

| PORCENTAJES (%) | |
|-----------------|--------------|
| Servicio CE VLL | 18,66 |
| Servicio IPNG | 17,1 |
| PROMEDIO | 17,88 |

Con los datos mostrados en la Tabla 4 se puede ver que en general, el software de configuración automática para servicios CE e IPNG en la red de MPLS Huawei de Internexa Colombia solo emplea menos de un cuarto (1/4) del tiempo con respecto a la configuración manual, lo cual lo hace muy eficiente para algunas de las operaciones que se realizan dentro de la compañía.

De acuerdo con lo anterior, el software en general representa una reducción de tiempos del 82.12%.

5.2 Experiencia de Usuario

Otro parámetro para medir el impacto del software en la compañía es la Experiencia de Usuario o también conocida como UX, ya que permite obtener una percepción del usuario que no solo depende del diseño del sistema sino también aspectos relacionados con las emociones.

Para conocer la percepción de los Ingenieros de Configuración en cuanto al software, se realizó una breve encuesta donde se realizaban preguntas como el número de veces que lo usan, la facilidad de su uso y beneficios que este ofrece en las operaciones diarias de la compañía.

De acuerdo con las opiniones de los Ingenieros de Configuración, ellos utilizan el software más de 15 veces a la semana, que según los tiempos mostrados en la sección 5.1 la configuración automática solo tomaría en total 1 hora 12 minutos (01:12), mientras que sin el programa se hubieran tomado 6 horas (06:00) para la configuración de estos 15 servicios.

Además, ellos expresaron que la herramienta es muy fácil de usar, muestra salidas de resultados entendibles para el usuario y que reduce la posibilidad de cometer errores.

5.3 Escalabilidad

Otra gran ventaja que brinda el software para la compañía es que éste es escalable, brindando la posibilidad en un futuro de configurar automáticamente dispositivos en topologías de red ubicadas en otros países como Perú, Argentina y Chile, proveer desconexión de servicios, integrar equipos L2-L3 de otras marcas, seleccionar una ruta en específico, así como brindar mecanismos para el control de tráfico.

5.4 Mejoras en los servicios entregados

Otro importante impacto en el uso del software desarrollado tiene que ver con la percepción de los clientes frente al servicio ofrecido por la organización. La principal razón que llevó a la modificación del antiguo software bajo el nuevo esquema de configuración basado en subinterfaces, es que los clientes se quejaban frecuentemente de que existían inconsistencias entre el ancho de banda real y el contratado frente a los servicios multimedia. En este contexto, la versión actual del software resuelve estas discrepancias gracias al uso de subinterfaces, permitiendo con esto consolidar la credibilidad y confiabilidad en la relación cliente/proveedor, y evitando posibles situaciones de orden legal, financiero o de reputación relacionadas con el incumplimiento de los acuerdos de nivel de servicio (SLA - Service Level Agreement, contrato técnico que define el tipo de servicio adquirido, especificando métricas a monitorear para garantizar la calidad del servicio, y así nivelar las expectativas de cumplimiento de ambas partes).

6. Conclusiones

- Haber tenido la oportunidad de realizar el semestre de industria en Internexa fue muy provechoso a nivel profesional porque aprendí como es el funcionamiento de las Telecomunicaciones a nivel comercial. Con el proyecto se conoció como se puede interconectar al país usando servicios CE e IPNG.
- Los proyectos de automatización para tareas operativas dentro de las compañías cada vez van tomando fuerza debido a la reducción de tiempos para dedicarse a otras labores y la disminución de errores humanos. El desarrollo de este software es el primer paso para los proyectos de automatización que tiene la compañía en su portafolio.
- La utilización de buenas prácticas de programación en Python y de los estándares definidos por entidades como IEEE, ISO o IETF son importantes en cualquier proyecto de desarrollo ya que facilitará el trabajo a las demás personas que van a estar involucradas en el proyecto.
- El trabajo en equipo y la comunicación asertiva con los Ingenieros de Configuración fueron esenciales para identificar la nueva problemática y encontrar soluciones para la misma.
- EL software para configuración automática de servicios CE e IPNG en la red MPLS Huawei de Internexa Colombia se considera óptimo ya que toma menos de un cuarto (1/4) del tiempo comparado con la configuración manual.
- A la hora de desarrollar software también es importante entregar una buena experiencia de usuario con el fin de generar credibilidad y placer al usuario a la hora de interactuar con el programa. La facilidad de la interfaz gráfica, la información que esta entrega y la velocidad de los procesos fueron determinantes para que la compañía lo aprobara.

7. Referencias Bibliográficas

- [1] Cloudflare Inc, "What is the OSI Model?," [En línea]. Disponible en: <https://www.cloudflare.com/learning/ddos/glossary/open-systems-interconnection-model-osi/> [Último acceso: 26 Mayo 2021].
- [2] A. Tanenbaum y D. Wetherall, Redes de Computadoras, Quinta ed., Pearson, 2012.
- [3] Imperva, "What is OSI Model | 7 Layers Explained | Imperva," [En línea]. Available: <https://www.imperva.com/learn/application-security/osi-model/>. [Último acceso: 26 Mayo 2021].
- [4] T. Noergaard, Embedded Systems Architecture: A Comprehensive Guide for Engineers and Programmers, Segunda ed., Elseiver, 2005.
- [5] Huawei Technologies Co, "Overview of MPLS," [En línea]. Available: <https://support.huawei.com/enterprise/en/doc/EDOC1000178173/953f01ce/overview-of-mpls>. [Último acceso: 26 Mayo 2021].
- [6] E. Rosen, A. Viswanathan y R. Callon, "rfc3031," Enero 2001. [En línea]. Available: <https://tools.ietf.org/html/rfc3031#section-2>. [Último acceso: 26 Mayo 2021].
- [7] L. Rosencrance, "What is a MultiProtocol Label Switching (MPLS)?," [En línea]. Available: <https://searchnetworking.techtarget.com/definition/Multiprotocol-Label-Switching-MPLS>. [Último acceso: 26 Mayo 2021].
- [8] J. Barberá, "MPLS: Una arquitectura de backbone para la Internet del siglo XXI," 22 Noviembre 2007. [En línea]. Available: <https://www.rediris.es/difusion/publicaciones/boletin/53/enfoque1.html>. [Último acceso: 26 Mayo 2021].
- [9] J. Montoya y C. Pérez, *Comunicación Personal*, Medellín, 2021.
- [10] Huawei Technologies Co, "Switches Agile de la serie S12700," [En línea]. Available: <https://e.huawei.com/es/products/enterprise-networking/switches/campus-switches/s12700>. [Último acceso: 26 Mayo 2021].
- [11] Huawei Technologies Co, "Huawei S12700 Series Agile Switches Brochure," 14 Mayo 2021. [En línea]. Available: <https://e.huawei.com/es/material/networking/campusswitch/b6e7f8ebe75c4440bbe3c5198ebeecf0>. [Último acceso: 26 Mayo 2021].
- [12] Huawei Technologies Co, "Huawei S12700 Series Agile Switches Datasheet," 14 Mayo 2021. [En línea]. Available:

<https://e.huawei.com/en/material/networking/campusswitch/ba8d3031f41e4e68a552d5e6b328face>. [Último acceso: 26 Mayo 2021].

- [13] Huawei Technologies Co, "S6700 series Datasheet (Detailed)," 10 Octubre 2015. [En línea]. Available: <https://www.router-switch.com/media/upload/product-pdf/huawei-s6700-series-switches-datasheet.pdf>. [Último acceso: 26 Mayo 2021].
- [14] Huawei Technologies Co, "Huawei S9300 Switch Product Brochures," [En línea]. Available:
<https://carrier.huawei.com/~media/CNBG/Downloads/Product/Fixed%20Network/carrierip-dcswitches/Huawei-S9300-Series-Switches-Product-Brochures-en.pdf>. [Último acceso: 26 Mayo 2021].
- [15] J. Hawkins, "Ethernet. vs. Carrier Ethernet: La nueva party line de redes," 14 Junio 2016. [En línea]. Available:
https://www.ciena.com.mx/insights/articles/Ethernet-vs-Carrier-Ethernet-The-New-Network-Party-Line-es_LA.html. [Último acceso: 26 Mayo 2021].
- [16] Ciena Corporation, "Qué es Carrier Ethernet?," [En línea]. Available:
https://www.ciena.com.mx/insights/what-is/What-is-Carrier-Ethernet_es_LA.html. [Último acceso: 26 Mayo 2021].
- [17] L. Rodríguez, "Redes de proxima generacion," 2013 Enero. [En línea]. Available:
https://www.researchgate.net/publication/259312180_Redes_de_proxima_generacion. [Último acceso: 26 Mayo 2021].
- [18] Huawei Technologies Co, "What Is a VLAN?," [En línea]. Available:
<https://support.huawei.com/enterprise/es/doc/EDOC1100086556>. [Último acceso: 26 Mayo 2021].
- [19] Huawei Technologies Co, "Configuring a Sub-interface," [En línea]. Available:
<https://support.huawei.com/enterprise/en/doc/EDOC1000142060/c44464f5/configuring-a-sub-interface>. [Último acceso: 26 Mayo 2021].
- [20] R. Belloso, Introducción a la Gestión de Redes, de *Planificación y Gestión de Red*, Primera ed., Maracaibo, 2010.
- [21] Hostinger Internacional, Ltd, "¿Cómo funciona el SSH?," 21 Mayo 2021. [En línea]. Available: <https://www.hostinger.es/tutoriales/que-es-ssh>. [Último acceso: 26 Mayo 2021].

- [22] M. Medina, *Diseño de una arquitectura de sistemas de información para la administración del alineamiento a estándares académicos*, Cartago, Cartago, 2018.
- [23] R. Puerta, "Análisis, Diseño e Implementación de una Aplicación Web mediante Java para la gestión de favoritos en Spotify," Junio 2015. [En línea]. Available:
http://oa.upm.es/44381/2/TFM_RAUL_PUERTA_SANCHEZ_MANUEL_GERARDO_A_CEVEDO_COELHO.pdf. [Último acceso: 26 Mayo 2021].
- [24] Oracle, "MySQL Database Service," [En línea]. Available:
<https://www.oracle.com/co/mysql/>. [Último acceso: 28 Mayo 2021].
- [25] Python, "Interfaces gráficas de usuario con Tk," [En línea]. Available:
<https://docs.python.org/es/3/library/tk.html>. [Último acceso: 28 Mayo 2021].
- [26] R. Pointer y F. Jeff, "Paramiko," [En línea]. Available:
<https://github.com/paramiko/paramiko>. [Último acceso: 28 Mayo 2021].