Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería Mecánica Eléctrica

DESCRIPCIÓN DEL FUNCIONAMIENTO Y SOLUCIÓN DE FALLAS DE UN CONTROLADOR DE SESIÓN DE FRONTERA, EN UNA RED DE TELEFONÍA DE VOZ SOBRE IP

Firmado electrónicamente por: José Francisco Goméz Rivera
Motivo: Informe final TESIS
Fecha: 23/04/2024 13:31:30
Lugar: Facultad de Ingeniería, USAC.

Alvaro Antonio Chacón Gómez

Asesorado por el Ing. Carlos Eduardo Guzmán Salazar

Guatemala, abril de 2024

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA

FACULTAD DE INGENIERÍA

DESCRIPCIÓN DEL FUNCIONAMIENTO Y SOLUCIÓN DE FALLAS DE UN CONTROLADOR DE SESIÓN DE FRONTERA, EN UNA RED DE TELEFONÍA DE VOZ SOBRE IP

TRABAJO DE GRADUACIÓN

PRESENTADO A LA JUNTA DIRECTIVA DE LA
FACULTAD DE INGENIERÍA
POR

ALVARO ANTONIO CHACÓN GÓMEZ
ASESORADO POR EL ING. CARLOS EDUARDO GUZMÁN SALAZAR

AL CONFERÍRSELE EL TÍTULO DE

INGENIERO ELECTRÓNICO

GUATEMALA, ABRIL DE 2024

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA

FACULTAD DE INGENIERÍA

NÓMINA DE JUNTA DIRECTIVA

DECANO	Ing. José Francisco Gómez Rivera (a. i.)
VOCAL II	Ing. Mario Renato Escobedo Martínez
VOCAL III	Ing. José Milton de León Bran
VOCAL IV	Ing. Kevin Armando Vladimir Cruz Lorente
VOCAL V	Ing. Fernando José Paz González
SECRETARIO	Ing. Hugo Humberto Rivera Pérez

TRIBUNAL QUE PRACTICÓ EL EXAMEN GENERAL PRIVADO

DECANO	Ing. José Francisco Gómez Rivera (a. i.)
EXAMINADOR	Ing. Byron Odilio Arrivillaga Méndez
EXAMINADOR	Ing. José Antonio de León Escobar
EXAMINADOR	Ing. Jorge Gilberto González Padilla
SECRETARIO	Ing. Hugo Humberto Rivera Pérez

HONORABLE TRIBUNAL EXAMINADOR

En cumplimiento con los preceptos que establece la ley de la Universidad de San Carlos de Guatemala, presento a su consideración mi trabajo de graduación titulado:

DESCRIPCIÓN DEL FUNCIONAMIENTO Y SOLUCIÓN DE FALLAS DE UN CONTROLADOR DE SESIÓN DE FRONTERA, EN UNA RED DE TELEFONÍA DE VOZ SOBRE IP

Tema que me fuera asignado por la Dirección de la Escuela de Ingeniería Mecánica Eléctrica con fecha 1 de junio de 2021.

Alvaro Antonio Chacón Gómez

Guatemala, 18 de junio 2023

Ingeniero
Julio César Solares Peñate
Coordinador Área de Electrónica
Escuela de Ingeniería Mecánica Eléctrica
Facultad de Ingeniería
Universidad de San Carlos de Guatemala

Estimado ingeniero Solares:

Hago de su conocimiento que he concluido la revisión del trabajo de graduación del estudiante de la carrera de ingeniería electrónica Alvaro Antonio Chacón Gómez, titulado: **DESCRIPCIÓN DEL FUNCIONAMIENTO Y SOLUCIÓN DE FALLAS DE UN CONTROLADOR DE SESIÓN DE FRONTERA, EN UNA RED DE TELEFONÍA DE VOZ SOBRE IP**

En calidad de ASESOR, doy mi APROBACIÓN al mismo, pues cumple con los objetivos que se propusieron para su elaboración.

Quedo en la mejor disposición de atender cualquier consulta sobre el trabajo del estudiante Chacón Gómez.

Atentamente,

Carlos Guzmán Salazar
ASESOR

CARLOS GUZMAN SALAZAR
Ingeniero Electricista
Col. No.2762

UNIVERSIDAD DE SAN CARLOS
DE GUATEMALA

FACULTAD DE INGENIERIA

Guatemala, 28 de junio de 2023

Señor director
Armando Alonso Rivera Carrillo
Escuela de Ingeniería Mecánica Eléctrica
Facultad de Ingeniería, USAC

Estimado Señor director:

Por este medio me permito dar aprobación al Trabajo de Graduación titulado: **DESCRIPCIÓN DEL FUNCIONAMIENTO Y SOLUCIÓN DE FALLAS DE UN CONTROLADOR DE SESIÓN DE FRONTERA, EN UNA RED DE TELEFONÍA DE VOZ SOBRE IP**, desarrollado por el estudiante **Alvaro Antonio Chacón Gómez**, ya que considero que cumple con los requisitos establecidos.

Sin otro particular, aprovecho la oportunidad para saludarlo.

Atentamente,

ID Y ENSEÑAD A TODOS

Ing. Julio César Solares Peñate
Coordinador de Electrónica

ESCUELA DE INGENIERÍA MECÁNICA ELÉCTRICA
FACULTAD DE INGENIERÍA
UNIVERSIDAD DE SAN CARLOS DE GUATEMALA

SIST.LNG.DIRECTOR.2.EIME.2024

El Director de la Escuela de Ingeniería Mecánica Eléctrica de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer el dictamen del Asesor, con el Visto Bueno del Coordinador de Área, al trabajo de Graduación del estudiante Alvaro Antonio Chacon Gomez: DESCRIPCIÓN DEL FUNCIONAMIENTO Y SOLUCIÓN DE FALLAS DE UN CONTROLADOR DE SESIÓN DE FRONTERA, EN UNA RED DE TELEFONÍA DE VOZ SOBRE IP, procede a la autorización del mismo.

Ingeniero Armando Alonso Rivera Carrillo
Director
Escuela de Ingeniería Mecánica Eléctrica

Guatemala, abril de 2024

Decanato
Facultad e Ingeniería

24189101- 24189102

LNG.DECANATO.OIE.172.2024

El Decano de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer la aprobación por parte del Director de la Escuela de Ingeniería Mecánica Eléctrica, al Trabajo de Graduación titulado: **DESCRIPCIÓN DEL FUNCIONAMIENTO Y SOLUCIÓN DE FALLAS DE UN CONTROLADOR DE SESIÓN DE FRONTERA, EN UNA RED DE TELEFONÍA DE VOZ SOBRE IP**, presentado por: **Alvaro Antonio Chacon Gomez** después de haber culminado las revisiones previas bajo la responsabilidad de las instancias correspondientes, autoriza la impresión del mismo.

IMPRÍMASE:

Firmado electrónicamente por: José Francisco Goméz Rivera
Motivo: Informe final TESIS
Fecha: 23/04/2024 13:29:34
Lugar: Facultad de Ingeniería, USAC.

Ing. José Francisco Gómez Rivera
Decano a.i.

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
DECANO a.i.
Facultad de Ingeniería

Guatemala, abril de 2024

ACTO QUE DEDICO A:

Dios	Por todas las bendiciones que he recibido en mi vida y por darme la fortaleza para cumplir esta meta. No hay nada imposible para Dios si confiamos en Él.
Mi madre	Carmen Gómez Sierra, por todo su amor, su dedicación, su esfuerzo, sus desvelos, por ser el motor que me impulsa cada día. Por ser ejemplo de humildad y bondad en mi vida.
Mi padre	Marco Antonio Chacón Fuentes, por su amor y su apoyo incondicional, por creer siempre en mí, por ser mi ejemplo de fortaleza, valentía, lucha y carácter frente a los obstáculos.
Mis hermanos	Mariela, Hevelia, Edgar y Ana Lucrecia Chacón Gómez, por todos los momentos que gracias a Dios hemos compartido y su apoyo incondicional.
Mis sobrinos	María José, Marco Antonio, Sara, José, Mateo, Ángel, Lucía, por todas las alegrías que hemos compartido y para quienes deseo ser un ejemplo de profesional y ser humano.

AGRADECIMIENTOS A:

Universidad de San Carlos de Guatemala	Por ser mi casa de estudios, y la institución que me permitió iniciar mi carrera profesional de ingeniería.
Facultad de Ingeniería	Por brindarme todas las herramientas y enseñanzas necesarias para convertirme en un profesional.
Mis amigos	Por brindarme su apoyo en cada uno de los cursos de la carrera.
Mi asesor Ingeniero Carlos Guzmán	Por su apoyo y asesoría en este trabajo de graduación.
M.Sc. Ingeniera Yulissa Córdova	Por acompañarme durante esta etapa, por sus consejos, su ejemplo de profesional y por motivarme a ser una mejor persona.

ÍNDICE GENERAL

ÍNDICE DE ILUSTRACIONES

FIGURAS

LISTA DE SÍMBOLOS

Símbolo	Significado
Hz	Hertz o ciclos por segundo

GLOSARIO

Decoder	Es un dispositivo utilizado para decodificar mensajes o señales enviadas en código, por ejemplo, las señales de televisión de un satélite.
Encoder	Define el movimiento en una señal eléctrica que puede ser leída por algún tipo de dispositivo de control en un sistema de control de movimiento, tal como un mostrador o PLC.
Kbps	Kilobits por segundo.
SBC	Controlador de sesiones en frontera, por sus siglas en inglés.

RESUMEN

Las telecomunicaciones han sido de vital importancia a través de los años, estas han venido evolucionando de la mano de nuevas tecnologías en varios campos, los medios de transmisión, la reducción de los dispositivos, entre otros aspectos. Las empresas hoy en día buscan maneras más agiles de comunicarse con sus clientes, así mismo los operadores de servicio buscan la manera de satisfacer estas necesidades. El uso de la telefonía pública conmutada aún es utilizado para suplir las necesidades de comunicación, sin embargo, la voz sobre IP cada día es más requerida por las empresas para agilizar sus comunicaciones.

Pero el uso del Internet conlleva tener una mayor seguridad en las redes, dado que puede haber personas mal intencionadas que utilicen las redes poco seguras para su beneficio. Esto no es solo preocupación de los grandes proveedores de servicios sino de cualquier empresa que tenga una conexión de voz sobre IP, pues debe tener especial cuidado con las amenazas. Es por esto que las empresas desarrolladoras de dispositivos han implementado el controlador de sesiones en frontera o SBC, por sus siglas en inglés, este elemento de la red dispone de funcionalidades capaces de brindar la seguridad que las empresas y los proveedores de servicios requieren.

En el presente informe de graduación se presentan conceptos básicos para comprender el funcionamiento de este dispositivo, así como la base para que los futuros ingenieros puedan realizar la operación y mantenimiento de este nodo dentro de una red, ya sea en un proveedor de servicios o dentro de una empresa.

OBJETIVOS

General

Describir el funcionamiento general de un controlador de sesión de frontera, utilizando el protocolo SIP para observar cómo aporta a la seguridad de la red de telefonía.

Específicos

1. Dotar a los profesionales relacionados a la rama de telecomunicaciones de una herramienta técnica de información, que complemente el conocimiento sobre el funcionamiento de la telefonía sobre IP.

2. Documentar los protocolos, procedimientos y funcionamiento del controlador de sesiones de frontera utilizando el protocolo SIP.

3. Proveer las bases y elementos de red que componen un sistema de telecomunicaciones.

4. Dar a conocer el protocolo SIP y su aplicación en un controlador de sesión de frontera.

5. Exponer casos prácticos de fallas y su solución en una red de telefonía IP.

INTRODUCCIÓN

A medida que las comunicaciones han avanzado, uno de los principales retos para los proveedores de servicios y también para cualquier empresa en general es la seguridad.

Desde hace algún tiempo la voz puede ser transportada por medio del protocolo de Internet que se conoce como voz sobre IP o como VoIP, en este método de transporte se implementaron protocolos, codificadores, elementos de red que soportan esta tecnología, entre otros.

Así mismo, dado que las empresas requerían la protección de sus datos, se desarrollaron elementos que, al introducirse a la red, ya sea del proveedor de servicios o a una empresa, brindaban la seguridad necesaria para que ninguna persona mal intencionada pueda hacer uso de la red para su beneficio personal.

1. CAPÍTULO 1

1.1. Digitalización de la voz

Digitalizar es el proceso en el cual una señal análoga se convierte a digital, es decir, a pulsos eléctricos que equivalen a dígitos combinados (0 y 1). Ampliando el concepto es posible decir que cualquier mensaje que pueda ser transformado a una señal eléctrica y codificarse puede ser almacenado o transmitido como un tren de impulsos a través de una red.

Las señales análogas son continuas en el tiempo, al ser transformadas a digitales se convierten en señales discretas, esto significa que solo pueden tomar valores enteros. La voz es una señal análoga continua en el tiempo, para ser transmitida en redes de datos se requiere tenerla codificada en formato digital. Al proceso de convertir la voz análoga a digital se le llama digitalización de la voz.

1.1.1. Señales análogas

Este tipo de señales se define puntualmente como aquellas cuya característica es que son continuas en el tiempo, por lo general se encuentran en la naturaleza.

1.1.2. Señales digitales

Son señales de valores discretos, es decir que toman valores enteros, por lo regular utilizan la lógica binaria, es decir unos y ceros, representando los valores altos o bajos de tensión eléctrica. Se debe tener en consideración tres

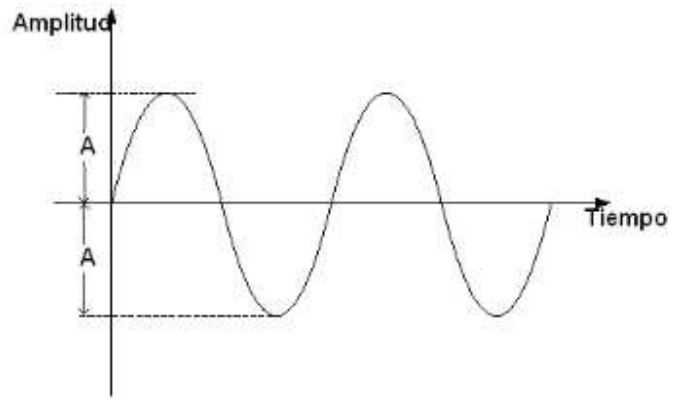
características importantes de las señales, mismas que se indican a continuación.

1.1.2.1. Amplitud

La primera característica a tener en cuenta es la amplitud de una señal, que se refiere al valor máximo que esta tenga. Es la distancia máxima entre el punto más alejado de una onda y el punto medio.

Figura 1.

Amplitud



Nota. Diagrama de señal de amplitud. Elaboración propia, realizado con Microsoft Word y Paint.

1.1.2.2. Frecuencia

El segundo concepto que debe definirse es la frecuencia de una señal, que se refiere a la cantidad de ciclos que pueden darse en un segundo. Se mide en ciclos por segundo o Hertz (Hz).

1.1.2.3. Fase

La fase de una señal se refiere al desplazamiento hacia la derecha o izquierda con respecto a un punto de referencia. Puede medirse en grados o radianes.

Dado que la voz es de carácter análogo y se mueve en forma de ondas, es importante su digitalización, ya que garantiza una mejor calidad, provee una mayor capacidad y trabaja con mayores distancias.

Para realizar este proceso de digitalización de la voz existen diversos métodos, se listan los más utilizados.

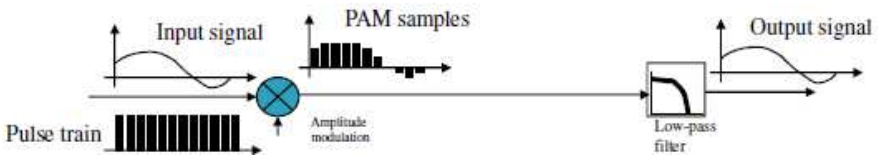
1.2. Modulación por amplitud de pulso

Pulse Amplitude Modulation, o PAM, establece un conjunto de tiempos discretos en los que la señal análoga es muestreada. La fase y la frecuencia permanecen estables y la amplitud es la que varía.

Esta modulación tiene como deficiencia que, aunque transforme la forma actual de la onda a una serie de pulsos, estos siguen teniendo la amplitud en señal analógica y no digital.

Figura 2.

Amplitud en señal analógica y no digital



Nota. Diagrama de amplitud en señal analógica y no digital. Elaboración propia, realizado con Microsoft Word y Paint.

1.3. Modulación por código de pulso

Una de las técnicas más empleadas es la modulación por código de pulso, o PCM (*Pulse Code Modulation*). Este proceso se basa en tres partes:

- Muestreo
- Cuantificación
- Codificación

1.3.1. Muestreo

Proceso mediante el cual una señal análoga se transforma a una serie de impulsos, estos impulsos toman el nombre de muestras.

Si se desea transmitir una señal de frecuencia f de un lugar a otro, según la teoría de la información, no es necesario transmitir la señal completa, sino que se puede transmitir muestras de esta señal tomadas a una velocidad del doble de la frecuencia máxima de la señal. A este método de toma de datos se le conoce como método de muestreo.

El número de muestras por unidad de tiempo que se toman de una señal se conoce como tasa o frecuencia de muestreo. Generalmente se expresa en hercios.

El oído humano percibe frecuencias hasta alrededor de los 20 Khz, según el teorema de muestreo una frecuencia de muestreo de 40 Khz sería la adecuada para digitalizar y poder reconstruir la señal.

Lo que resulta de muestrear una señal es un tren de impulsos, cada uno con un valor igual al valor instantáneo que tenía la señal al momento del muestreo. El muestreo real se toma durante cierto tiempo y no en un instante.

En la teoría mediante el teorema de muestreo las señales de voz que van entre los 300 y 3400 Hz han de muestrearse a una frecuencia igual o superior a los 6800 Hz, en la práctica se toma una frecuencia de muestreo de 8000 Hz, esto equivale a 8000 muestras por segundo.

1.3.2. Cuantificación

Al asignarle valores discretos a las amplitudes provenientes del muestreo se está cuantificando las muestras, posterior a la cuantificación estas muestras ya son digitales.

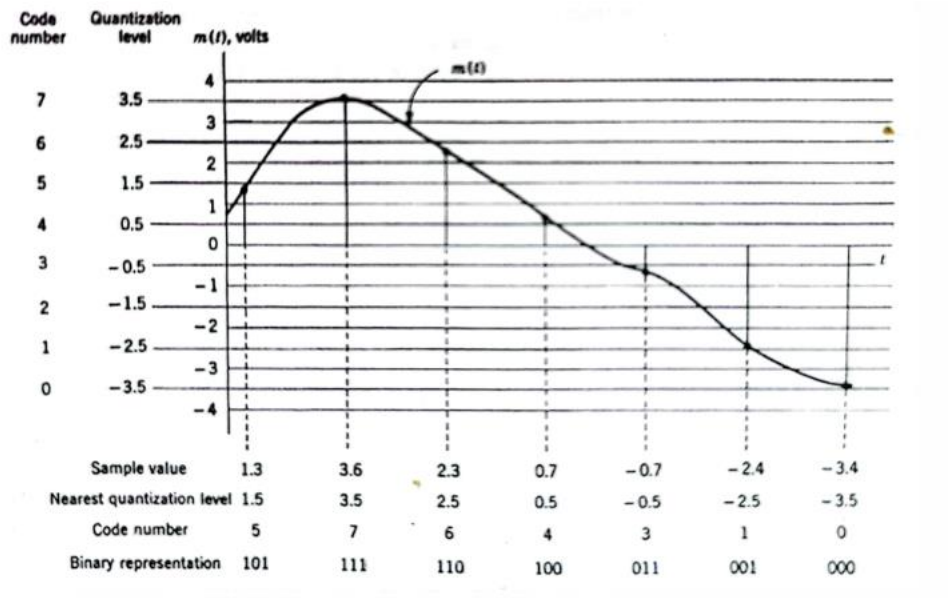
La amplitud de las muestras puede tomar valores finitos que van desde el 0 hasta el valor más alto de la señal muestreada. Al asignar un valor a varias amplitudes que están en un rango definido se están cuantificando las muestras. A cada intervalo en que se divide el rango de amplitudes se le llama intervalo de cuantificación.

1.3.3. Codificación

Al tener ya las muestras cuantificadas el objetivo se centra en desarrollar métodos de transmisión que proporcionen una mejor calidad de audio, menor complejidad, entre otros aspectos. Si la señal cuantificada se representa como una sucesión de unos y ceros se está codificando la señal para poder ser transmitida en forma de paquetes.

Figura 3.

Codificación de señal



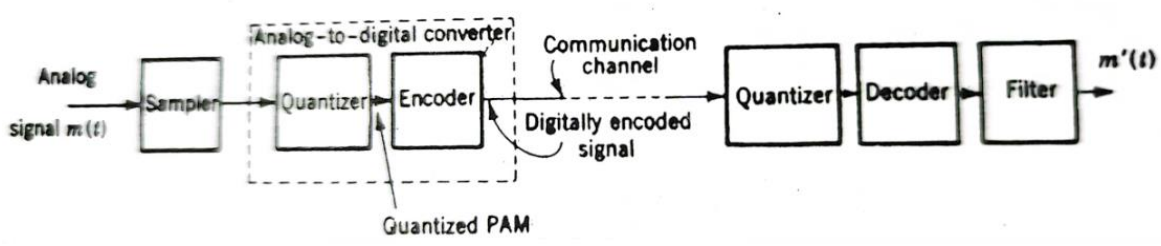
Nota. Datos para codificación de señal. Obtenido de E. Rodger y W. Tranter (2014). *Principles of comunications. Systems, Modulation and Noise.* (p. 135.) Wiley.

El sistema PCM se compone del *encoder,* el cual convierte una señal análoga en una serie de palabras digitales codificadas, esto se conoce como un convertidor análogo a digital A/D. Al llegar la señal digital desde el canal de

comunicación lo que se realiza es la operación inversa, para esto se utiliza el *decoder,* el cual convierte la señal digital a una secuencia de pulsos cuantizados.

Figura 4.

Sistema PCM



Nota. Datos del sistema PCM. Obtenido de E. Rodger y W. Tranter (2014). *Principles of comunications. Systems, Modulation and Noise.* (p. 137.) Wiley.

1.4. Ancho de banda

El ancho de banda se define como la cantidad de información que fluye en una red en un periodo de tiempo determinado. En una red generalmente se denota como N bits por segundo. De acuerdo a esto se tiene entonces miles de bits por segundo kbps, millones de bits por segundo Mbps, miles de millobes de bits por segundo Gbps o billones de bits por segundo Tbps.

Para señales analógicas se puede definir el ancho de banda en función de la cantidad de espectro magnético que ocupa la señal. Se denota como ciclos por segundo o hercios (Hz).

1.5. Redes de datos

El ser humano tiene la necesidad de comunicarse con sus iguales, esto ha sido una de las causas para la evolución del cómo trasladar la información.

Pero, ¿qué es una red de comunicación? Se puede definir como un conjunto de elementos con características comunes que se interconectan entre sí por un medio físico, con el fin de transmitir la información de un punto a otro.

Con el nacimiento de las computadoras, trasladar la información se hacía por medio de unidades de almacenamiento magnético, los disquetes. Este proceso era bastante engorroso.

A principios de la década de los 80´s las redes de datos se incluyeron para poder trasladar la información, pero cada empresa o institución hacía uso de sus propios estándares, esto provocó que la comunicación entre redes no fuera del todo práctica.

Hoy en día las redes de datos juegan un papel vital, ya que permiten la transmisión de información y la comunicación mundial. Y es que la forma en que se vive actualmente ha evolucionado, ya no solo es necesario que las empresas se comuniquen entre sí, la tecnología ha evolucionado y la mayoría de personas posee un dispositivo inteligente con el cual puede estar conectado con el resto del mundo.

Debido a la necesidad del ser humano para comunicarse de una forma rápida y eficaz, fueron incorporándose medios para que esto pudiera darse.

La infraestructura en la que se montan las comunicaciones puede variar en términos de:

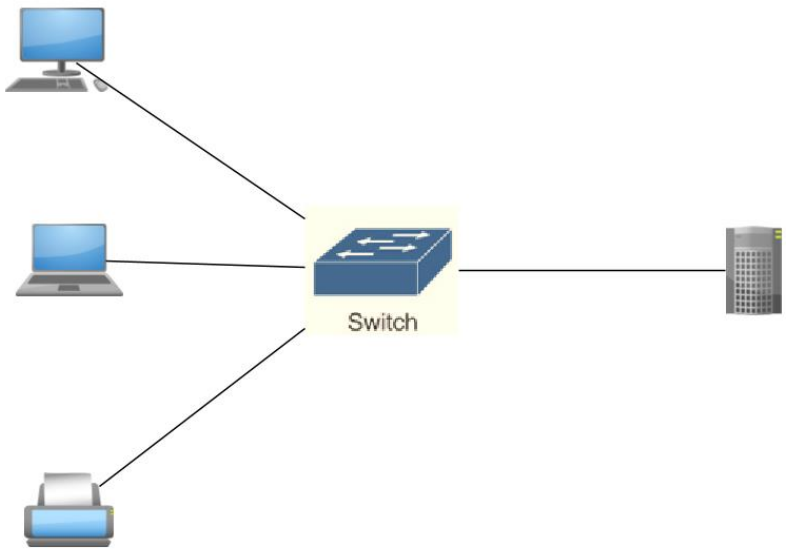
- El tamaño de su área de cobertura
- La cantidad de usuarios conectados
- La cantidad y el tipo de servicios

1.5.1. Red de Área Local (LAN)

Este tipo de red fue desarrollada para compartir información localmente dentro de una institución. Tiende a estar administrada por una única organización. La seguridad y el acceso están implementados en el nivel de red.

Figura 5.

Red de Área Local (LAN)



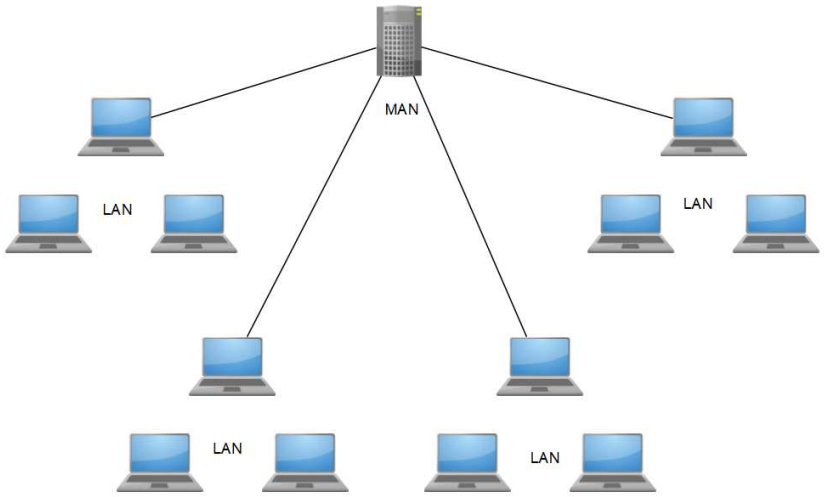
Nota. Diseño de Red de Área Local. Elaboración propia, realizado con yEd Graph Editor.

1.5.2. Red de Área Metropolitana (MAN)

Una red metropolitana interconecta dispositivos que se ubican en diversos edificios que pertenecen a una misma institución. Representa una evaluación del área local, ya que abarca un espacio físico más amplio, es decir tiene una mayor cobertura.

Figura 6.

Red de Área Metropolitana (MAN)



Nota. Diseño de Red de Área Metropolitana. Elaboración propia, realizado con yEd Graph Editor.

1.5.3. Red WAN

La función principal de una red WAN es la de interconectar las redes LAN. Brinda la conectividad entre zonas extensas geográficamente. Entre sus características están:

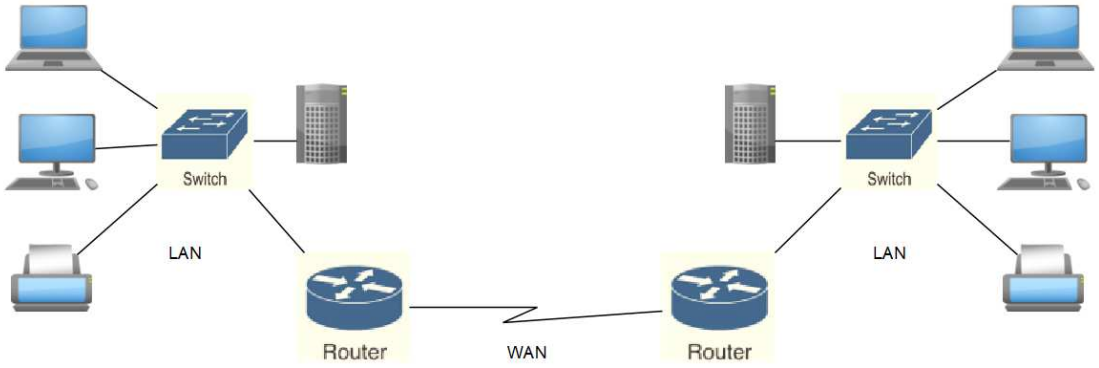
- Diseñada para operar en áreas extensas y distantes.

- Interconecta servidores en tiempo real para compartir información.
- Habilita el uso de servicios corporativos de mayor demanda como correo electrónico y transferencia de archivos.

Las redes WAN hacen uso de dispositivos de red diseñados para interconectar las redes LAN. Se puede apreciar la importancia de estos dispositivos, la configuración, instalación y el mantenimiento son funciones que no se pueden dejar de lado.

Figura 7.

Red WAN



Nota. Diseño de Red WAN. Elaboración propia, realizado con yEd Graph Editor.

1.6. Dispositivos que conforman una red

En el sexto apartado de este primer capítulo se debe aclarar que los dispositivos se pueden categorizar en dos grupos: dispositivos de usuario y dispositivos de red.

1.6.1. Dispositivos de usuario

Estos son los encargados de conectar a los usuarios a la red, se conocen comúnmente como *hosts.* Permiten la transmisión de la información a través de la red. Una de las cualidades de los *hosts* es que pueden existir sin necesidad de la red, sin embargo, quedarán aislados del mundo, limitados a uso personal.

En cada *host* existe la circuitería electrónica para poder conectarse a la red, se conoce como tarjeta de red.

Entre los dispositivos de usuario están los siguientes:

- Computadoras personales
- Impresoras
- Teléfonos IP

1.6.2. Dispositivos de red

Encargados de conectar los dispositivos de usuario a la red, con el fin de transportar la información que cada usuario comparta. Hay diversos dispositivos de red y cada uno cumple una función específica de acuerdo a su diseño, conversión de formatos, conexión de cableado y transferencia de datos. Hay por ejemplo algunos de estos dispositivos:

- Repetidor: es un dispositivo que reconstruye una señal atenuada o distorsionada, ya sea por impurezas o por ruido en el medio.

- Hub: permiten concentrar distintos puntos de conexión para ser tratados como un mismo elemento de red. Transmiten un mensaje que se conoce

como *broadcast*, significa que repite el paquete de datos que recibe en el número de puertos activos y que están conectados a él.

- Puente: estos dispositivos conectan redes LAN, su trabajo es verificar si los datos recibidos pertenecen o no a la red destino.

- *Switches:* administran de forma inteligente una red LAN, poseen la capacidad de determinar si los datos pertenecen o no al segmento de red al cual están configurados.

- *Router:* este dispositivo reúne las características de los elementos de red listados anteriormente, regenera señales, concentra múltiples redes y posee la lógica necesaria para identificar si los paquetes pertenecen a la red destino, o de lo contrario desecharlos.

1.7. Protocolos de comunicación

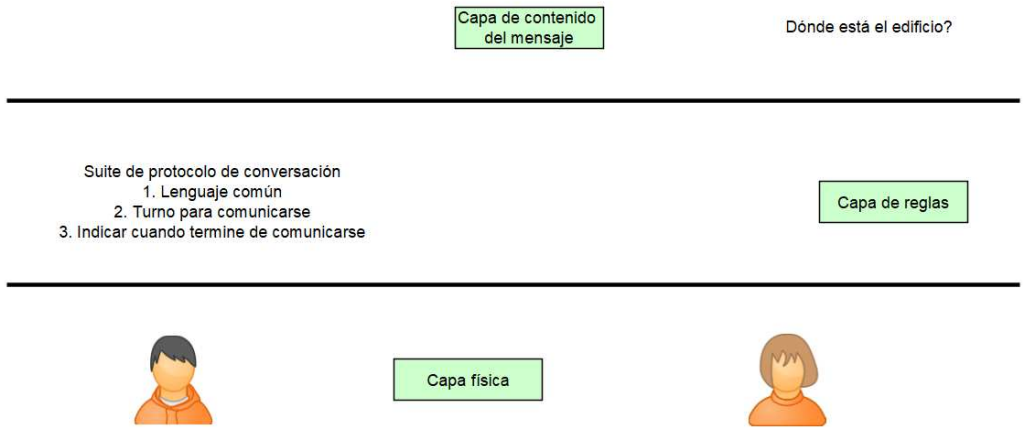
Las reglas que rigen la comunicación por una red son conocidas como protocolos. Estos protocolos son específicos de las características que tenga la comunicación. Al comunicarse verbalmente con otra persona las reglas que se utilizan son muy diferentes a las reglas que se utilizarán, por ejemplo, para enviar una carta (Netacad, 2005).

Los protocolos de comunicación dictan los parámetros que determinan cuál será la semántica y la sintaxis que deben emplearse en el proceso de la comunicación. Esto permite también que, si en algún caso se pierden los datos, estos puedan ser recuperados.

Para lograr una comunicación exitosa entre los *host* de una red es necesaria la interacción de una gran cantidad de protocolos distintos. Al conjunto de protocolos interrelacionados se les denomina suite de protocolos, los cuales están implementados sobre el *hardware* y *software* de cada *host* y dispositivo de red.

Figura 8.

Protocolos de comunicación



Nota. Suite de protocolos. Elaboración propia, realizado con yEd Graph Editor.

Para lograr exitosamente la interacción entre los protocolos se diseñaron modelos de capas para comprender la estructura de cada red, los modelos sirven para representar el funcionamiento de la red y por eso son llamados modelos de referencia.

Los protocolos más importantes en una red son TCP, *Transmission Control Protocol* o Protocolo de Control de la Transmisión, y el IP o *Internet*

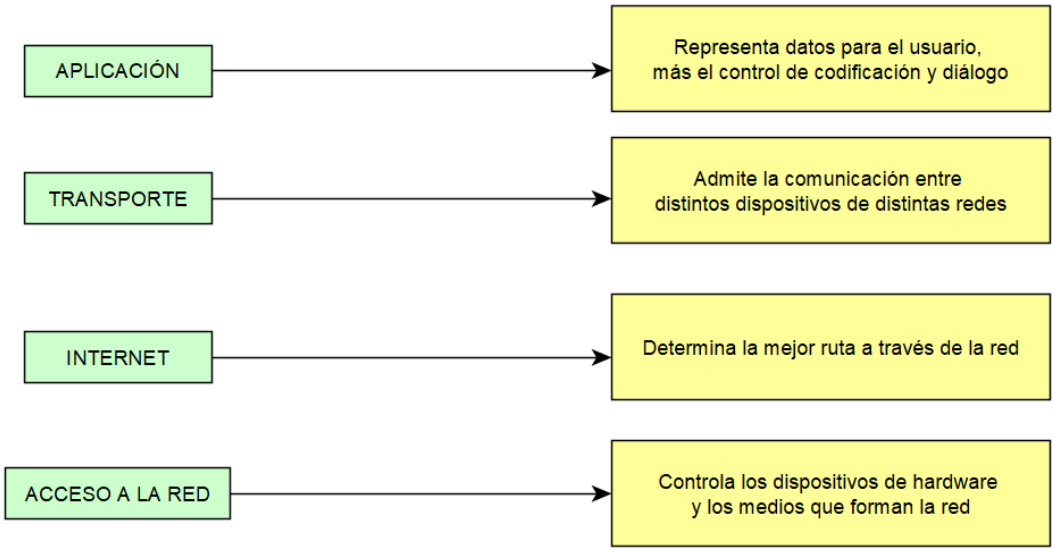
Protocol, es decir Protocolo de Internet. En conjunto posibilitan el enlace entre todos los equipos que acceden a la red.

1.7.1. Modelo TCP/IP

Modelo de capas desarrollado por el Departamento de Defensa de los Estados Unidos, también se conoce como modelo de Internet. Este está orientado a la comunicación entre redes de extremo a extremo. Brinda el formato, transmisión, enrutamiento de los datos enviados y cómo debe recibirlos el destino.

Figura 9.

Modelo cuatro capas: aplicación, transporte, Internet y acceso a la red



Nota. Diagrama de modelo cuatro capas: aplicación, transporte, Internet y acceso a la red. Elaboración propia, realizado con Curso CCNA Exploration versión 4.0.

El protocolo de Internet IP es el que se encarga de determinar la transferencia de cada datagrama dentro de la red IP.

1.7.2. Modelo OSI

Protocolo creado por la ISO (International Organization for Standardization), significa *Open Systems Interconexion Reference Model*, o Modelo de Referencia de Interconexión de Sistemas Abiertos. Este protocolo permitió la separación entre las operaciones necesarias para el funcionamiento de una red.

Esta separación hizo que fuera más entendible, los fabricantes pudieron especializarse en ciertas áreas y, quizás lo más importante, dio un marco de referencia para la resolución de problemas. Este modelo en la actualidad es utilizado con fines académicos.

El modelo OSI divide en siete capas el proceso de transmisión de la información, cada capa se encarga de ejecutar una parte en todo el proceso.

- Aplicación: es la capa que está en contacto con el usuario. Provee de acceso a la red a las aplicaciones.

- Presentación: su función es la de convertir la información en un formato genérico, también brinda codificación, compresión y encriptación de los datos.

- Sesión: inicia y finaliza sesiones.

- Transporte: en esta capa se elige el tipo de transmisión, confiable con el protocolo TCP o no confiable con el protocolo UDP, así como el número de puerto.

- Red: destinada a proveer el direccionamiento Lógico (IP) y a encontrar la mejor ruta hacia el destino.

- Enlace: destinada a proveer el direccionamiento Físico (dirección MAC).

- Física: en ella se contempla la electrónica y medios físicos, ya sean eléctricos, ópticos, de radiación, entre otros, que son utilizados para el funcionamiento de la red.

1.8. Red de telefonía

Los avances tecnológicos se han dado a través de los años por la necesidad que tiene el ser humano para resolver problemas. La comunicación es quizás uno de los más importantes. Es importante conocer las bases para poder llegar a entender las nuevas tecnologías.

Una red de telefonía interconecta dos o más usuarios para establecer una comunicación por medio de la voz. En un principio eran redes punto a punto, después fueron avanzando a dispositivos electromecánicos, llegando a elementos digitales.

1.8.1. Topología de la red telefónica

El diseño de las redes se basa en la optimización de los recursos, decidir entre un tipo u otro depende de factores como:

- La cantidad de usuarios. Se tiene que tomar en cuenta cuántos usuarios utilizarán los recursos.

- Ubicación geográfica. Esto es importante ya que depende de la geografía el tipo de recursos que se deben implementar.

- La escalabilidad de la red. Esto es muy importante, ya que se debe tener en cuenta que los usuarios van a ir en aumento y será necesario incluir o ampliar los elementos de la red para dar el servicio.

La configuración en forma de árbol es la más empleada para la interconexión de centrales, una red tipo malla se utiliza para conectar directamente determinados nodos del árbol. En esta configuración mixta se optimiza la jerarquización.

1.8.2. PSTN

Significa Red Telefónica Pública Conmutada, es una red global de conmutación de circuitos tradicional, diseñada principalmente para transmitir voz en tiempo real.

La infraestructura necesaria para lograr una llamada exitosa sobre una PSTN consta de:

- Codificación de la voz: la voz, al llegar a la PSTN, se convierte a digital, se transporta y al otro extremo se decodifica para tener de nuevo la voz original.

- *Switches* PSTN: mueven el tráfico entre los enlaces y proveen los circuitos y las conexiones necesarias para el manejo de llamadas.

- PBX: es una pequeña central telefónica, usualmente se encuentra en empresas que requieren más que una línea residencial.

- Señalización: los protocolos de comunicación que sirven para dirigir el flujo de la llamada.

- Dispositivos: pueden ser análogos, como los que se tienen en las residencias, o digitales, empleados usualmente en las PBX.

1.8.3. NGN

Red de siguiente generación (NGN, *Next Generation Networking,* por sus siglas en inglés), es la evolución de las redes monolíticas a redes convergentes. Mezcla las arquitecturas PSTN tradicionales y las basadas en conmutación de paquetes.

Utiliza protocolos de control entre los que se encuentran el H.248, MGCP, H.323 y SIP, y de transporte entre los que se encuentran el RTP, UDP e IP.

1.8.3.1. Arquitectura de red NGN

La red interconecta diferentes elementos, va a depender del servicio que brindará. Cada elemento se interconectará por medio de interfaces y protocolos hacia el núcleo de la red.

1.8.3.2. Principales características de la NGN

Arquitectura abierta: brinda la facilidad de incorporarse con sistemas de aprovisionamiento de terceros. Soporta un control distribuido, así como seguridad y protección.

Aprovisionamiento independiente: el servicio del aprovisionamiento debe estar separado de la operación de la red, utilizando mecanismos de control.

Multiplicidad: ofrece la flexibilidad de soportar múltiples tecnologías de acceso.

1.8.4. IMS

Subsistema multimedia IP, o por sus siglas en inglés *IP Multimedia Subsystem,* originalmente estandarizada por la 3GPP como parte de la versión 5 del UMTS. Es una nueva infraestructura de red móvil, una nueva manera de brindar servicios de comunicación en tiempo real, tanto para usuarios finales como para empresas.

El IMS describe la arquitectura para soportar telefonía y servicios multimedia a través del enrutamiento de paquetes por medio de direcciones IP. Esto habilita a la red a incorporar servicios de voz, multimedia y datos en una sola plataforma. Solo se requiere que los equipos utilicen el protocolo SIP, el cual permite la señalización de sesiones.

Para lograr entender la arquitectura del IMS este se divide en capas:

- Capa de acceso: soporta cualquier tipo de acceso de alta velocidad, puede ser acceso móvil (VoLTE) o acceso WiFI (VoWiFI).

- Capa de transporte: define el ruteo a nivel IP de cada paquete que utiliza la arquitectura.

- Capa de control: capa en la cual se toma el control de la señalización y su interacción con los servidores de las aplicaciones.

- Capa de aplicación: la conforman los servidores de aplicación, son los responsables de integrar la totalidad de los servicios, funcionalidades y conversión de protocolos.

El protocolo SIP pasa a ser preponderante, sin embargo, aparecen protocolos como Diameter.

2. CAPÍTULO 2

2.1. Voz sobre IP (VoIP)

En estos días, en vez de utilizar la tradicional red conmutada, se utiliza una red IP como el Internet para transmitir llamadas.

Voz sobre IP o *Voice over Internet Protocol* (VoIP) utiliza la red IP para conversaciones entre teléfonos. Esto es posible ya que cuando se habla por medio de un micrófono la voz genera señales eléctricas dentro del dispositivo, estas señales son analógicas, el voltaje puede tener cualquier valor dentro de un rango. Esta señal analógica es convertida a una señal digital usando algún algoritmo que está implementado en el dispositivo utilizado, pudiendo ser un teléfono VoIP o un *softphone,* el cual es un software instalado en un computador que funciona como teléfono. La voz ya digitalizada es arreglada en paquetes y enviada a la red IP.

Estos paquetes, comúnmente llamados datos, son enrutados a través de servidores y puertas de acceso hasta el destino. Una vez en el destino la voz vuelve a ser convertida a señal análoga para poder escucharla.

Durante este proceso se utilizan protocolos, como por ejemplo SIP, para controlar la llamada. RTP sirve para mantener la calidad de servicio y la legitimidad de la transmisión de paquetes y SDP para la media.

2.2. Introducción a SIP

En los últimos años el protocolo de inicio de sesión o SIP, por sus siglas en inglés *session initiation protocol*, ha pasado a ser un estándar para los servicios de telefonía y multimedia en las redes móviles y fijas. SIP fue diseñado con la visión de revolucionar la forma en que los servicios de comunicación eran desarrollados y operados.

En SIP una sesión se establece por medio de diálogos del tipo Request/Response o solicitud/respuesta, cada paquete de SIP tiene una estructura que se compone de encabezado y cuerpo, el cuerpo suele ser del protocolo SDP. Para la señalización se utiliza SIP, y para la media por lo general se utiliza RTP.

2.2.1. IETF

Para entender de dónde proviene el protocolo SIP se debe conocer la organización que en cierta manera lo rige, la IETF, por sus siglas en inglés *Internet Engineering Task Force*, juega el papel de organización para los estándares del Internet.

IETF es una gran comunidad abierta internacional de diseñadores de redes, operadores, proveedores e investigadores preocupados por la evolución de la arquitectura y el buen funcionamiento de Internet.

A mediados de los años noventa la IETF implementaba diferentes protocolos necesarios para los servicios que se basaban en telefonía IP. El RTP: *Real-Time Protocol*, el SDP: *Session Description Protocol* y el SAP: *Session Announcement Protocol* (Handley, Perkins y Whelan, 2013).

2.2.2. Protocolos utilizados en SIP

En cuanto a este tema, importante por su relación con todo lo anterior, debe saberse que, ya que SIP es conformado por varios protocolos, a continuación se observan los más utilizados.

2.2.2.1. Protocolo RTP

Protocolo de transferencia en tiempo real, o RTP por sus siglas en inglés *Real-time transfer Protocol*, proporciona servicios de entrega de extremo a extremo para datos, ya sea audio o video, con características en tiempo real. En sus orígenes fue diseñado para soportar conferencias múltiples de multimedia, pero en la actualidad ha sido usado para diversas aplicaciones (Schulzrinne, 2003).

Por ejemplo, al querer escuchar una canción, primer se debe descargar, en este escenario no interesa tanto la velocidad de descarga, sino más bien que se descargue de manera correcta. Pero ¿qué pasaría si en lugar de descargarla se quisiera solo escuchar? Y se quisiera tener no solo toda la canción, sino que también la velocidad a la cual se está descargando, de lo contrario la canción no se escucharía igual. En este caso se necesita una transmisión en tiempo real.

Sin embargo, el protocolo RTP no asegura la entrega a tiempo ni la garantía de la calidad de servicio. Para esto se basa en servicios de capa inferior como UDP o TCP. RTP brinda una funcionalidad para transportar contenido en tiempo real.

El protocolo se divide en dos partes: una, la que lleva datos en tiempo real conocida como RTP o *Real Time Protocol,* y la que monitorea la calidad de

servicio y transmite la información conocida como *Real Time Control Protocol* o RTCP.

2.2.2.2. Protocolo SDP

El Protocolo de Descripción de Sesión, o SDP *Session Description Protocol*, por sus siglas en inglés, es utilizado para describir las sesiones multimedia en un formato que sea entendido por todos los participantes en una red. Según como sea esta descripción una parte decide si unirse o no y cuándo y cómo unirse a una sesión (Schulzrinne, 2006).

Su principal propósito es transmitir información sobre la media en sesiones para ayudar a los participantes a unirse o recopilar información sobre una sesión en particular. Este protocolo incluye:

- El nombre de la sesión y su propósito.
- El tiempo que la sesión está activa.
- Los medios que componen la sesión.
- Información para recibir los medios, como por ejemplo las direcciones, puertos, formatos, entre otros.
- El tipo de media (video, audio, entre otros).
- El protocolo de transporte (RTP/UDP/IP, entre otros).
- El formato de la media.

SDP es una descripción textual estructurada, transmite el nombre y el propósito de la sesión, los medios, protocolos, formatos de códec, temporización e información del transporte. Un participante en la comunicación verifica esta información y decide si unirse o no a la sesión.

El protocolo consiste en un número de líneas de texto en la siguiente forma:

<tipo> = <valor>

Donde el tipo define un parámetro de sesión único y el valor proporciona un valor específico para ese parámetro. A continuación, se muestran algunos parámetros del protocolo divididos por tipo:

2.2.2.2.1. Descripción de la sesión

v = versión del protocolo (Ej. 0 = Origin)
o = propietario/creador he identificador de sesión
o = <nombre de usuario> <id de la sesión> <versión> <tipo de red (Ej. IN = internet)> <tipo de dirección> <dirección>
s = nombre de la sesión
i = información de la sesión, parámetro opcional
u = URI (*Universal Resource Identifier*) de descripción
e = dirección de correo
p = número de teléfono
c = información de conexión
c = <tipo de red> <tipo de dirección> <dirección de conexión>
b = información del ancho de banda
b = <modificador> : <valor de ancho de banda>

z = ajustes de la zona horaria

k = llave de encriptación

a = cero o más líneas de atributo de la sesión.

2.2.2.2.2. Descripción del tiempo

t = tiempo que la sesión está activa

t = <tiempo de inicio> <tiempo de fin>

r = cero o más veces de repetición

2.2.2.2.3. Descripción del medio

m = nombre del medio y dirección de transporte

m = <media (audio, video, aplicación, data, control)> <puerto> <transporte> <lista fmt>

i = título del medio

c = información de la conexión

b = información del ancho de banda

k = llave de encriptación

a = cero o más atributos del medio

A continuación, se presenta un ejemplo de una descripción de SDP tomado de la RFC2327:

v=0

o=mhandley 2890844526 2890842807 IN IP4 126.16.64.4

s=SDP Seminar

i=A Seminar on the session description protocol

u=http://www.cs.ucl.ac.uk/staff/M.Handley/sdp.03.ps

e=mjh@isi.edu (Mark Handley)
c=IN IP4 224.2.17.12/127
t=2873397496 2873404696
a=recvonly
m=audio 49170 RTP/AVP 0
m=video 51372 RTP/AVP 31
m=application 32416 udp wb
a=orient:portrait

2.2.2.3. Protocolo SAP

SAP difunde periódicamente un paquete de anuncios a una dirección y puerto *multicast.* Este anuncio es *multicast* y tiene el alcance de la sesión que está anunciando, asegurando que los destinatarios del anuncio estén dentro del alcance de la sesión que describe el anuncio. Mantiene locales los anuncios de la sesión local. De esta manera un oyente de SAP se enterará de que todas las sesiones están siendo anunciadas, lo que permite unirse a esas sesiones.

En esos días el proceso para establecer una llamada VoIP entre dos usuarios basados en los estándares antes mencionados requería que un llamante iniciara su aplicación de audio y video con cierta IP y puerto, llamaba a el receptor por medio del teléfono o enviaba un correo informándole sobre qué IP, puerto y tipo de compresión estaba utilizando. El receptor entonces encendía su propia aplicación de audio y video e informaba al llamador sobre su IP y puerto. Este método era viable para algunos investigadores para comunicarse a larga distancia, pero claramente no era aplicable para los usuarios normales de Internet.

2.2.2.4. Protocolo SIP

El Protocolo de Inicio de Sesion (SIP – *Session Initiation Protocol*) fue un intento de la comunidad de IETF por proveer un protocolo de señalización que no solo permitiera llamadas por teléfono, sino que al mismo tiempo fuera usado para iniciar cualquier clase de sesiones de comunicación. Dado esto, SIP puede ser usado para llamadas VoIP, así como para configurar una sesión de juego o controlar un dispositivo como una refrigeradora. Los estándares de SIP están definidos en la RFC 3261 por la Internet Engineering Task Force (IETF), la cual es una comunidad internacional de diseñadores de redes, operadores, interesados en la evolución de la arquitectura del Internet y su operación.

En estos días el protocolo de inicio de sesión se encuentra en muchas plataformas, ya que brinda una facilidad para establecer y mantener sesiones. SIP es soportado por prácticamente cualquier proveedor de teléfonos, PBX y es parte del IMS (*IP Multimedia Subsystem*). SIP es utilizado para localizar, negociar y establecer sesiones de cualquier tipo, como VoIP, video, juegos, texto, control de llamadas, entre otros. Una sesión de comunicación puede involucrar cualquier dispositivo como computadoras, teléfonos móviles, teléfonos IP, entre otros (CISCO, s.f.).

SIP se basa en una serie de protocolos existentes y utiliza un lenguaje basado en texto, esto significa que los mensajes de SIP son fáciles de programar e interpretar, haciendo mucho más fácil el uso de diferentes proveedores.

Las especificaciones del protocolo de inicio de sesión describen tres tipos de componentes: usuarios, o como se le conoce por sus siglas en inglés: UA (*user agents*), *hosts* de red, también conocidos como *proxy servers*, y los servidores de registro. Los usuarios pueden ser aplicaciones de VoIP.

SIP es un protocolo de Internet de la capa de aplicación, que sirve para establecer, manipular y terminar sesiones de comunicación. Pero SIP no transporta la media en sí, esto es manejado por códecs, dentro de los programas de comunicación o dispositivos.

Una característica de SIP es la habilidad de usar la dirección de un usuario final como una única dirección pública que une todas las comunicaciones.

Por ejemplo, para el usuario Juan Pérez su dirección sería SIP:juanperez@empresa.com, utilizando esta dirección un usuario puede comunicarse con Juan Pérez en sus muchos dispositivos para comunicación, sin tener que saber todas las direcciones o números de teléfonos de Juan Pérez.

Para complementar esta dirección SIP provee el mecanismo llamado Identificador Uniforme de Recursos o URI (*Uniform Resource Identifier*), que establece un esquema común de direccionamiento para todos los dispositivos del usuario.

El formato de URI sigue la forma básica de una dirección web o una dirección de correo, siendo esta dirección-contacto@dominio. A continuación algunos ejemplos de una dirección URI:

Para un teléfono: sip:50212345678@empresa.com;user=phone
Para Fax: sip:50212345678@empresa.com;user=fax

Esencialmente SIP es utilizado para iniciar y finalizar sesiones de media. Como se había indicado, SIP se compone de varios protocolos, entre ellos el Protocolo de Descripción de Sesión (SDP – *Session Description Protocol*), el cual

contiene información sobre la sesión que se está configurando, como por ejemplo el tipo de medio, el *codec* a utilizar y el protocolo para transportar la media).

2.3. Componentes en una red SIP

En el tercer apartado de este segundo capítulo de la presente investigación, se hablará de los componentes para una red SIP, los cuales se presentan en los siguientes incisos.

2.3.1. Agentes de usuario

User Agents (UAs) son aplicaciones en los dispositivos SIP finales, como por ejemplo un teléfono SIP, un teléfono móvil, una computadora, entre otros, que intervienen entre el usuario y la red SIP.

Un UA puede actuar tanto como cliente o como servidor. Cuando el UA envía mensajes SIP está en modo UA Cliente (UAC) y cuando recibe mensajes actúa como UA Servidor (UAS)

2.3.2. SIP server

Los servidores SIP brindan información centralizada y habilitación de servicios en un ecosistema SIP. Para comprender esto es necesario considerar el siguiente contenido:

2.3.2.1. Los servidores de registro

Cuando los usuarios están en línea necesitan asegurarse que los otros usuarios sepan que están disponibles para realizar y hacer llamadas. El servidor

de registro autentica y registra usuarios cuando entran en línea, además de esto guarda información de la identidad lógica y del dispositivo que usa para la comunicación. Los dispositivos se identifican por su URI.

2.3.2.1.1. Servidor de localización

A medida que los usuarios se trasladan la red debe estar continuamente al tanto de sus ubicaciones, el servidor de localización es una base de datos que realiza un seguimiento de los usuarios y sus ubicaciones. La entrada de este servidor se obtiene desde el servidor de registro y proporciona información importante a los servidores proxy y de redirección. Un servidor proxy utiliza esta información para obtener la asignación de direcciones SIP lógicas a direcciones SIP físicas, de modo que las sesiones de comunicación puedan establecerse y mantenerse adecuadamente.

2.3.2.1.2. Servidor de redirección

Si los usuarios no están en su dominio de origen las sesiones que ellos poseen deben redirigirse hacia ellos. El servidor de redirección mapea una solicitud SIP destinada a un usuario a su URL más cercana. Por ejemplo, si una llamada es para ejemplo@empresa.com y el usuario está en carretera, el servidor de redirección de su compañía puede responder hacia el usuario que hace la llamada o el servidor *proxy* con la dirección de contacto del teléfono celular del destino, así la llamada puede ser redireccionada hacia el teléfono celular de la persona.

2.3.2.1.3. Servidor Proxy

Un servidor *proxy* recibe las solicitudes SIP, las procesa y las transfiere hacia el destino mientras envía respuesta al origen. Un servidor *proxy* puede actuar tanto como servidor o como cliente y tiene la facilidad de modificar las solicitudes SIP antes de trasladarlas al destino. Un servidor *proxy* solo participa entre el inicio y fin de la sesión, una vez establecida la sesión la comunicación fluye directamente entre origen y destino.

2.3.2.1.4. Servidor de presencia

Para que los usuarios detecten la presencia de sus compañeros, para mejorar la comunicación, a menudo se utiliza un servidor de presencia, el cual acepta, almacena y distribuye la información de presencia. Hay dos tipos de usuarios en los servidores de presencia, los que producen la información de sí mismos, para que sea almacenada y distribuida, y los usuarios que ven esta información.

2.4. Sesión en SIP

En SIP una sesión se establece por medio de diálogos del tipo request / response, cada paquete de SIP tiene una estructura que se compone de encabezado y cuerpo, el cuerpo suele ser del protocolo SDP, para la señalización se utiliza SIP y para la media por lo general se utiliza RTP.

2.4.1. Estructura de mensajes SIP

Como ya se ha comentado, SIP es un protocolo basado en texto, y utiliza el grupo de caracteres UTF-8. Un mensaje SIP puede ser tanto una solicitud de

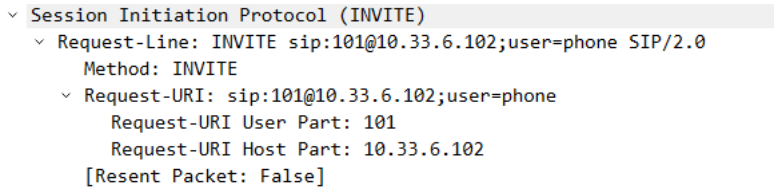
un cliente a un servidor o una respuesta del servidor al cliente. Ambos mensajes de solicitud utilizan el formato básico.

Ambos tipos de mensajes se conforman de una línea principal, uno o más campos de cabecera, una línea en blanco indicando el fin de los campos de cabecera y el cuerpo del mensaje.

A continuación, se presenta la estructura de los mensajes SIP:

Figura 10.

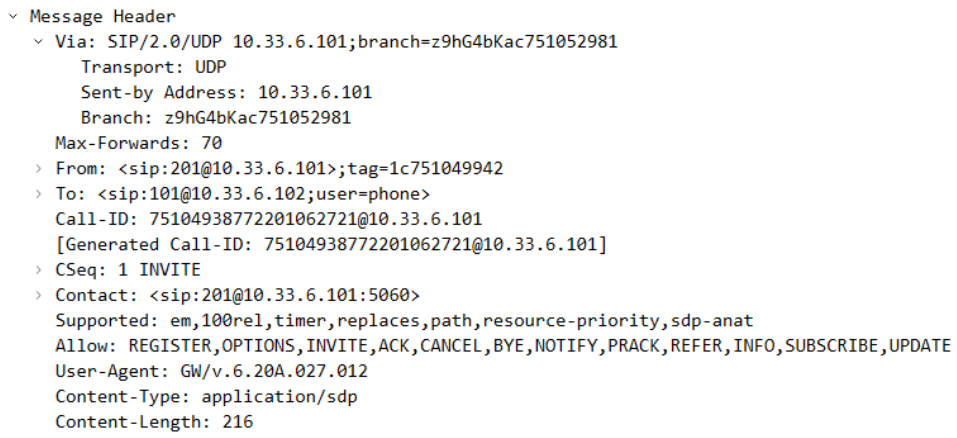
Request-line o línea de solicitud



Nota. Presentación de la estructura de los mensajes SIP. Elaboración propia, realizado con WireShark.

Figura 11.

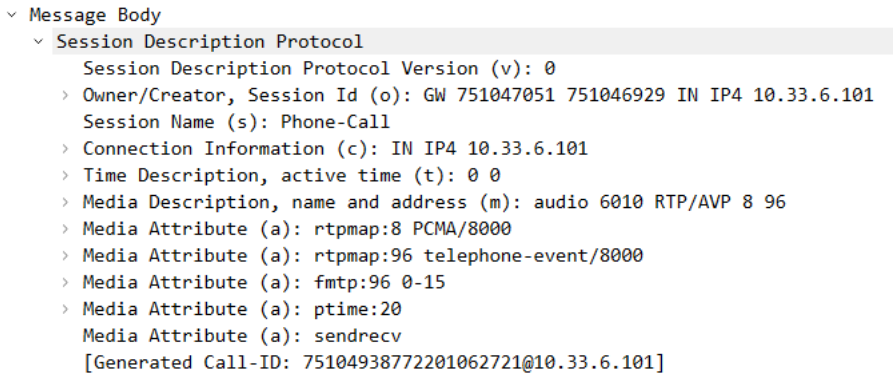
Message headers o mensajes de encabezado



Nota. Presentación de la estructura de los mensajes SIP. Elaboración propia, realizado con WireShark.

Figura 12.

Message body o cuerpo del mensaje



Nota. Presentación de la estructura de los mensajes SIP. Elaboración propia, realizado con WireShark.

La línea principal se compone de lo siguiente:

- Línea de solicitud / Línea de estado

2.4.1.1. Solicitudes

Las solicitudes SIP se diferencian, ya que contienen una línea de solicitud como línea principal. Una línea de solicitud contiene un nombre de método, una solicitud URI y la versión del protocolo separado solo por un espacio.

Formato de una línea de solicitud:

- Línea de solicitud = método solicitud-URI versión SIP

Nótense los espacios entre los campos, los cuales deben ir.

2.4.1.2. Métodos

Se definen seis métodos en el protocolo SIP, los cuales se enumeran a continuación:

- *Register:* para el registro de la información del contacto.
- *Invite, ack, cancel:* para configurar sesiones.
- *Bye:* para terminar sesiones.
- *Options:* para consultar a los servidores sobres sus capacidades.

2.5. Solicitud-URI (*Request-URI*)

Esta solicitud puede llegar como SIP o SIPS e indica el usuario o servicio hacia el cual se direcciona la solicitud. Esta solicitud-URI no debe contener espacios o caracteres de control y no debe encerrarse en los caracteres <>.

Los elementos deben soportar solicitudes-URI con otros esquemas además de SIP o SIPS, como por ejemplo “tel”, esto debe ser transformado por el elemento para llegar a SIP o SIPS URIs.

2.6. Versión SIP

Tanto los mensajes de solicitud como de respuesta deben incluir la versión SIP utilizada.

2.7. Respuestas

Las respuestas SIP se diferencian de las solicitudes por tener como línea principal una línea de estado. Una línea de estado contiene la versión del protocolo seguido de un número de estado de código y un texto asociado, cada elemento separado por un espacio.

Formato de línea de estado.

Línea de estado = Versión SIP Codigo_de_Estado Texto_de_Motivo

2.8. Código de estado

En el octavo apartado de este segundo capítulo se trata acerca del código de estado. Este es un entero formado por tres dígitos que indica el resultado de un intento de entender o satisfacer una solicitud.

2.9. Texto de motivo

Es un intento de dar una breve descripción del código de estado.

Regresando al código de estado, se sabe que el primer dígito define la clase de respuesta, los últimos dos no tienen una categorización. Dicho esto es posible inferir que los códigos entre 100 y 199 pertenecen a una respuesta 1xx, los códigos entre 200 y 299 pertenecen a una respuesta 2xx y así sucesivamente. Se tienen hasta seis valores para el primer dígito para la versión SIP 2.0, los cuales son:

- 1xx: provisional – solicitud recibida, continuación del proceso de la solicitud.

- 2xx: exitoso – la acción fue exitosamente recibida, entendida y aceptada.

- 3xx: redirección – una acción adicional debe ser atendida para completar la solicitud.

- 4xx: error de cliente – la solicitud contiene una mala sintaxis o no puede ser procesada por el servidor.

- 5xx: error de servidor – el servidor falló en cumplir una solicitud probablemente válida.

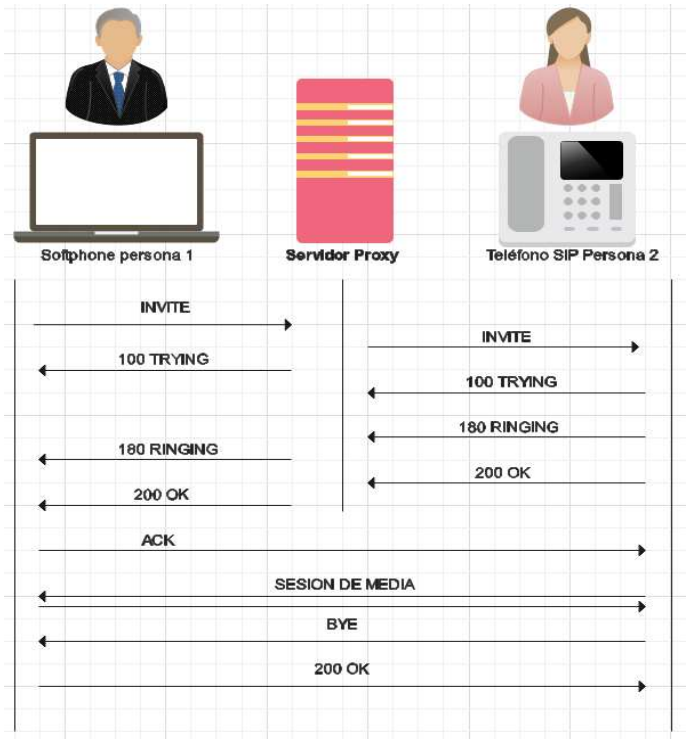
- 6xx: falla global – la solicitud no puede ser tramitada en ningún servidor.

2.10. Flujo de llamada en SIP

A continuación, se presenta cómo se establece una sesión en SIP:

Figura 13.

Flujo de llamada en SIP



Nota. Presentación de la estructura de una sesión SIP. Elaboración propia, realizado con WireShark.

En el escenario de la figura 13 se ve en diagrama de escalera una típica sesión SIP. Esta muestra dos usuarios, los cuales van a tener las siguientes direcciones:

Softphone persona 1: persona1@empresa1.com
Teléfono SIP persona 2: persona2@empresa2.com

Muestra también el servidor *proxy* que servirá para conectar los dos UAs. La comunicación sigue la siguiente lógica:

1. *Invite*:persona1@empresa1.com el UAC inicia la sesión invitando a persona2@empresa2.com. Una solicitud *Invite* es generada y enviada hacia persona2. Este primer mensaje *Invite* contiene los parámetros del protocolo SDP o protocolo de descripción de sesión, en el cual se define el tipo de media que el usuario que llama soporta y hacia dónde desea que la media sea enviada.

2. Un servidor DNS resuelve la dirección SIP hacia el servidor *proxy*, el cual se llamará proxy.empresa2.com, el mensaje *Invite* es enviado hacia el servidor *proxy.*

3. El servidor *proxy* recibe y procesa el mensaje de *Invite* y busca el contacto en el servidor de registro el contacto de persona 2.

4. 100 *trying:* un mensaje es enviado desde el servidor *proxy* hacia persona 1, indicando que está tratando de establecer la comunicación.

5. El servidor de registro devuelve la dirección host@phone.empresa2.com donde actualmente se encuentra la persona 2.

6. *Invite*: el servidor *proxy* genera y envía un segundo mensaje de *Invite* hacia el servidor host@phone.empresa2.com.

7. 180 Ringing: el UAS en host@phone.empresa2.com le pregunta a persona 2 si quiere aceptar la llamada, la persona 2 escucha un timbre.

8. 200 Ok: el mensaje de aceptación de la llamada es envía hacia el servidor *proxy.*

9. 200 Ok: el servidor *proxy* envía el mensaje de aceptación hacia persona 1.

10. ACK: la persona 1 responde al mensaje de aceptación con un ACK o *acknowledgement* que le indica al servidor *proxy* y a la persona 2 que está lista para empezar la llamada.

11. Sesion Media: la llamada se establece y las dos personas comienzan a comunicarse.

12. *Bye:* al final de la conversación la persona 2 cuelga la llamada. Se envía el mensaje de *bye* hacia la persona 1.

13. 200 Ok: la persona 1 responde al mensaje de *bye,* termina la sesión.

Como se observa, el flujo describe el inicio de una llamada, pero la flexibilidad de SIP es que funciona para cualquier otro establecimiento de sesión, no limitándose a llamadas por teléfono.

2.11. Codecs

Abreviación de codificador-decodificador. Su principal función es la de convertir una señal analógica a digital basándose en un algoritmo, pueden encontrarse tanto en hardware como en *software*.

Hay muchos tipos de *codecs,* cada uno con características distintas, entre ellas, el ancho de banda, tiempo de compresión y calidad de servicio (VOIP-INFO, 2005).

En VoIP los *codecs* se utilizan para reducir el ancho de banda y entre los más comunes están:

- G711, uno de los *codecs* más antiguos, orientado para trabajar en un entorno de redes de área local (LAN). Usa un sistema de modulación PCM con una velocidad de transmisión de 64 kbps. Se presenta en dos versiones la A-*law*, la cual es el estándar utilizado en Norteamérica y Japón, y el U-*law* presente en el resto del mundo, ambos trabajan con una forma de muestreo logarítmica.

- G722 hace uso de una técnica de modulación que divide la señal a muestrear, antes de digitalizarla, en dos bandas de frecuencia. Cada banda pasa por un proceso de muestreo, cuantificación y codificación. Opera en velocidades de transmisión en el orden de los 48, 56 y 64 kbps.

- G723.1 basado en un modelo de cuantificación por multipulso de máxima probabilidad y en un modelo de compresión que utiliza predicción lineal por análisis algebraico. Opera a una velocidad de transmisión de 5.3 y 6.3 kbps.

- G726 orientado a la red de telefonía pública PSTN, opera a velocidades de 16, 24, 32 o 40 kbps.

- G728 utiliza una taza de transferencia de 16kbps.

- G729 está orientado a trabajar en redes de área local, sin embargo utiliza un algoritmo muy complejo y demanda más recursos de memoria y CPU para trabajar, es por eso que fue desarrollada una versión más ligera: el G729, el cual se utiliza para VoIP.

3. CAPÍTULO 3

Las comunicaciones están en constante cambio, las empresas están reemplazando los sistemas convencionales por soluciones VoIP, servicios en la nube, para mejorar su productividad y reducir sus costos operativos.

Debido a esta transición entre los sistemas hacia nuevas y mejores tecnologías, las empresas deben implementar nuevos sistemas y prácticas para resguardar la infraestructura, proteger la comunicación y preservar la disponibilidad. Es acá donde nace la necesidad de tener un equipo en la red capaz de conectar a las empresas con la red pública de Internet, redes privadas u operadores de servicios, para manejar el tráfico, aplicar políticas, proveer seguridad y eficiencia a las comunicaciones.

Tal como se ve en capítulos anteriores, SIP es el protocolo dominante para las comunicaciones vía IP. La mayoría de proveedores de servicios brindan soluciones de troncales SIP, que brindan bajos costos, alternativa a los tradicionales circuitos E1. Soporta un gran rango de plataformas desde PBX hasta teléfonos inteligentes.

Trasladarse a una comunicación IP introduce varios riesgos en seguridad, incompatibilidad entre equipos, así como pone en riesgo la calidad de servicio.

Los dispositivos convencionales de una red IP, tales como *routers, firewalls,* entre otros, no están diseñados para manejar comunicaciones en tiempo real y no solucionan los problemas antes mencionados.

Un SBC procesa tráfico en tiempo real de protocolos de comunicación, como SIP, cabe resaltar que un SBC termina y vuelve a originar cada sesión de comunicaciones, lo que le permite inspeccionar el tráfico y aplicar un control y políticas bien delimitadas.

Las empresas utilizan los SBC para conectar y controlar el flujo de tráfico a través de una infraestructura de comunicación en tiempo real hacia el Internet, otras redes IP y hacia troncales SIP de proveedores de servicio.

Entonces, al mismo tiempo en que las empresas migran sus servicios a comunicaciones IP, se encuentra una manera para manejar y salvaguardar la información, no dejando de lado los niveles de calidad de servicio ofrecidos a los usuarios.

3.1. SBC más que un *firewall*

Es importante entender la diferencia fundamental entre un SBC, el cual está diseñado para manejar y controlar en tiempo real la voz y video de una sesión de comunicación, y un equipo como el *firewall* que se delimita a bloquear o permitir que los datos fluyan.

Una sesión IP en comunicaciones se compone de información de señalización, he información de media (voz o video). La señalización y la media viajan por medio de diferentes protocolos IP. Como ya se ha visto, SIP maneja el establecimiento y control de la sesión, y por ejemplo RTP es usado para entregar las tramas de audio o video.

La mayoría de los *firewall* solo ofrecen configuraciones básicas para el protocolo SIP, lista de control de acceso (ACL) que pueden permitir o denegar el

tráfico SIP. Los *firewall* no pueden manejar un control en tiempo real de las comunicaciones SIP de la misma manera que lo hacen los SBC.

Figura 14.

SBC_a

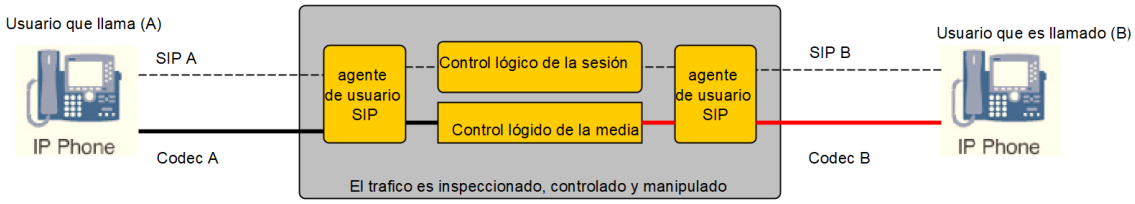
Nota. SBA_a. Elaboración propia, realizado con yEd Graph Editor.

La diferencia se basa en la arquitectura, un *firewall* SIP se implementa con un servidor SIP *proxy*, el cual es responsable para retransmitir y controlar la información de señalización SIP, no es diseñado para desenvolverse en un control de la media en RTP.

En cambio, los SBC son implementados como un agente de usuarios B2B, que procesa activamente ambas tramas, señalización y media. Un agente de usuarios B2B termina una sesión SIP (llamante) y establece una nueva sesión SIP (llamado). Esto permite al SBC inspeccionar y manipular el contenido de toda la sesión, afianzando las políticas de seguridad y manejando eficientemente la comunicación.

Figura 15.

Usuarios B2B



Nota. Usuarios B2B. Elaboración propia, realizado con yEd Graph Editor.

El SBC mantiene el estado de la sesión, controla y manipula la señalización SIP más la media RTP asociada. El SBC es muy versátil, dependiendo la ubicación en la red así serán las distintas funciones para los servicios, tales como: troncales SIP, servicios alojados, servicios en la nube, entre otros. El SBC delimita la red interna con la red del proveedor de servicios.

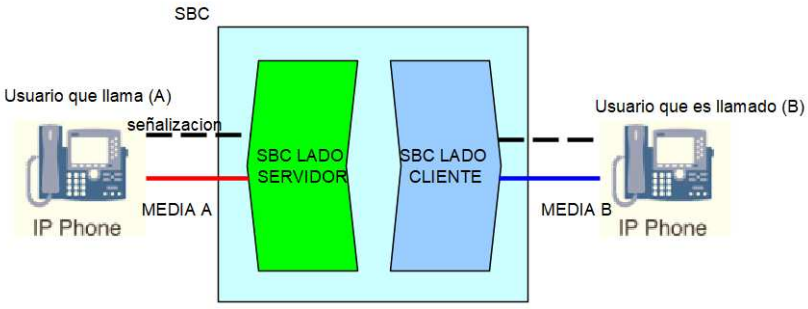
3.2. Funciones de un SBC

Ya que existen diferentes fabricantes, los SBC se presentan en distintas formas y son usados por los operadores y las empresas de distintas maneras para lograr el objetivo que se proponen. Sin embargo, se pueden diferenciar algunas formas de implementarlos.

Encontramos el SBC B2BUA, *back-to-back user agent*, se refiere a un agente de usuario espalda con espalda. Este se podría comparar con un servidor *proxy* que divide las transacciones SIP en dos partes: on lado que se dirige hacia el lado cliente y el otro lado hacia el lado servidor.

Figura 16.

SBC B2BUA



Nota. SBC B2BUA. Elaboración propia, realizado con yEd Graph Editor.

Mientras que un servidor *proxy* normal mantiene el estado de la información relacionada únicamente a las transacciones activas, un SBC B2BUA mantiene el estado de la información para llamadas activas y solo las elimina cuando la llamada es terminada.

El SBC B2BUA actúa como servidor para la parte que llama, mientras que para la parte a la que llaman actúa como cliente. Debido a esto el SBC termina la llamada que fue generada por la parte que llama, y empieza una nueva hacia la parte llamada. El mensaje *Invite* enviado por el SBC ya no contiene la referencia del número que llama. Ahora incluye los campos Via y Contact en el Header de sí mismo y no el origen. El SBC puede manipular la información listada en el campo Call-Id y en From.

En esta configuración el SBC puede manipular el tráfico de la media, cambiando la información en los campos *c* y *m* del SDP. En el *Invite* enviado por el SBC debido a que establece un nuevo diólogo, puede manipular los campos CSeq y Max-Forwards value.

3.3. Ocultación de topología

Como resultado del establecimiento de la sesión SIP, los nodos involucrados sabrán las direcciones IP desde donde reciben y envían el tráfico. Esto significa que si un usuario A móvil llama a un usuario B fijo vía SIP, el usuario sabrá la dirección IP del Gateway que le da salida al servicio fijo. En el campo VIA del encabezado irá la información de estas direcciones IP.

Esto presenta una brecha de seguridad, un usuario con malas intenciones puede utilizar esta información para atacar a los operadores *proxy* o incluso tener acceso al Gateway directo de las redes, provocando costos que el operador de servicio cobrará.

Para ocultar los nodos internos de un operador, todos los mensajes salientes de la red del operador viajarán a través del SBC. El SBC reemplazará las direcciones de los nodos internos por las de sí mismo. Los campos como Contact, Via y PAI incluirán únicamente la dirección propia del SBC. Esto se da también en la media, en el SDP.

3.4. El SBC como *proxy*

Un servidor *proxy* es aquel que toma la función de intermediario entre dos redes, es posible verlo de la siguiente manera, suponiendo que un usuario requiere acceso a un servidor web, pero en medio se tiene un servidor *proxy,* el usuario al hacer la petición de sesión la hará hacia el *proxy* este a su vez traslada la petición pero con su propia dirección hacia el servidor web, esto significa que el servidor web nunca sabrá que el usuario hizo la solicitud, ya que él únicamente verá al servidor *proxy.* El SBC actúa como un *proxy* tanto para la señalización como para la media.

A continuación, se presenta la función del SBC como *proxy* en cada elemento de la red.

- Terminal: el terminal usa la dirección del SBC como su dirección *proxy.*

- *Core:* desde el punto de vista del *core,* el SBC viene a ser el usuario.

- SBC: como *proxy* el SBC se encarga de la traducción de direcciones de sesión, el ancho de banda y el estado de la sesión para el terminal y el *core.*

3.5. NAT – Transversal

Por sus siglas en inglés *Network Address Translators,* o traductores de direcciones de red, es utilizado para superar la falta de direcciones IPv4, ocultando la red incluso de un operador en unas cuantas direcciones IP. Los dispositivos por atrás de NAT utilizan direcciones IP privadas que no pueden ser enrutadas en Internet.

En el caso de un usuario localizado por detrás de NAT, utilizará una red IP privada como su dirección de contacto, en los campos Contact y Via tanto para el encabezado como en el SDP. Esta información resultará inútil para cualquiera que quiera contactar al usuario desde el Internet.

Existen diferentes soluciones NAT como STUN he ICE, va a depender del comportamiento de NAT y del escenario. Cuando utilizamos el SBC para solventar los problemas de NAT el uso más común del SBC es actuar como interface pública para los usuarios. Esto se logra reemplazando la información de contacto con la del SBC.

Para que un usuario sea alcanzable a través de las interfaces públicas del SBC, el SBC manipulará la información de registración del usuario. El usuario incluirá su dirección IP privada y su información de contacto en las solicitudes de *Register.* Las llamadas hacia esta dirección fallarán, ya que no son enrutadas públicamente. Entonces el SBC reemplaza la información en el *contac* del encabezado con su propia dirección IP. Esta es la información que ahora es registrada en el servidor *register.* Las llamadas destinadas al usuario se direccionarán ahora al SBC. Pero ¿cómo sabrá el SBC a qué usuario se direcciona la llamada? El SBC guarda una copia local del registro de los usuarios, esta copia incluye las direcciones IP privadas, la SIP URI, así como la IP pública incluida en el encabezado que fue asignada al mensaje por NAT.

Como una alternativa, el SBC puede resguardar esta información en los mensajes SIP reenviados, cuando en la información de contacto del usuario se coloca un formato especial y se adiciona como parámetro al *contac* en el encabezado. La información de contacto puede incluir la dirección IP privada, la SIP URI, así como la dirección IP pública en el encabezado del mensaje SIP. Cuando el servidor de registro recibe una solicitud para el usuario retornará la información completa del contacto al *proxy,* que incluirá la información en el mensaje SIP. El SBC recibe esta información de solicitud y la usará para enrutarla hacia el usuario. Esto reduce el uso de recursos y memoria del SBC.

Aunque tener una copia del registro de los usuarios resguardada en el SBC consumirá recursos, esto puede tener una ventaja, ya que NAT solo guardará la información de la IP privada con la IP pública durante un corto periodo de tiempo. Para mantener esta información será necesario que se estén enviando mensajes *Register* constantemente en cortos periodos de tiempo. Con una base de datos de la información únicamente será necesario reenviar los mensajes

Register en largos periodos de tiempo (horas), reduciendo así los recursos dedicados para este proceso.

3.6. NAT transversal y media

Si para la señalización el proceso de NAT parece complicado, es un poco más complejo habilitar la media para que utilice NAT. El problema es el mismo: los dispositivos que utilicen NAT no podrán ser alcanzables desde redes externas.

Cuando se trata de la media, el SBC, en vez de enviar la media a la dirección IP y puerto que indica el SDP, lo envía a un usuario simétricamente hacia la dirección donde el usuario ha enviado su propia media.

Cabe resaltar que se deben utilizar los mismos puertos para enviar y recibir la media para que esto funcione.

3.7. Denegación de servicio y protección contra sobrecarga

Como cualquier otro servidor que se interconecta a la red, los servidores VoIP son propensos a ataques DOS.

Este tipo de ataque se puede confundir con tráfico VoIP normal que en ocasiones tiende a aumentar, y es difícil deducir. Es por eso que los operadores deben incorporar mecanismos de seguridad que monitoreen la carga y el tráfico entrante para identificar el aumento de carga y así prevenir una interrupción total del servicio.

Para detener este tráfico fraudulento y la sobrecarga de los procesadores, en los equipos VoIP el SBC incluye ofrece varias características de seguridad. Entre ellas se pueden encontrar:

- Limitación de tráfico: el operador puede limitar la taza de llamadas entrantes o registraciones, una vez la taza llega a su valor definido como crítico el SBC empezará a denegar las solicitudes entrantes.

- Lista negra dinámica: una lista negra estática usualmente es utilizada para bloquear todo el tráfico proveniente de ciertas fuentes sin primero procesarlo, es decir se colocan IP´s en una lista negra y todo el tráfico proveniente de esas IPs será bloqueado. Sin embargo, por lo general no es posible saber desde dónde se producirá un ataque, es por eso que el SBC monitorea el tráfico y si se cumplen ciertas características los orígenes de este tráfico malintencionado serán colocados en una lista negra dinámica. Estas características pueden ser el número de mensajes enviados por una fuente en un periodo de tiempo, el contenido del mensaje o si una fuente hace muchas llamadas hacia varios destinos en avalancha. Una vez que sea agregado a lista negra el tráfico será descartado.

- Filtrado de contenido: un atacante puede intentar tener acceso a la red enviando mensajes malformados en su contenido, analizando el contenido de los mensajes y rechazando cualquiera que tenga algún contenido malicioso el SBC puede proteger a la red que tiene a sus espaldas.

- Priorizacion de las llamadas: los clientes desean que incluso en un momento de ataque o sobrecarga, sus llamadas aún sean procesadas, es por eso que el SBC puede mantener en su base de datos usuarios

registrados, aceptando únicamente llamadas provenientes de estos usuarios registrados.

3.8. Regulación

Uno de los temas que no son ampliamente discutidos es lo concerniente a la regulación, la interceptación legal debe de incluir tanto la señalización como la media, y el SBC maneja ambas.

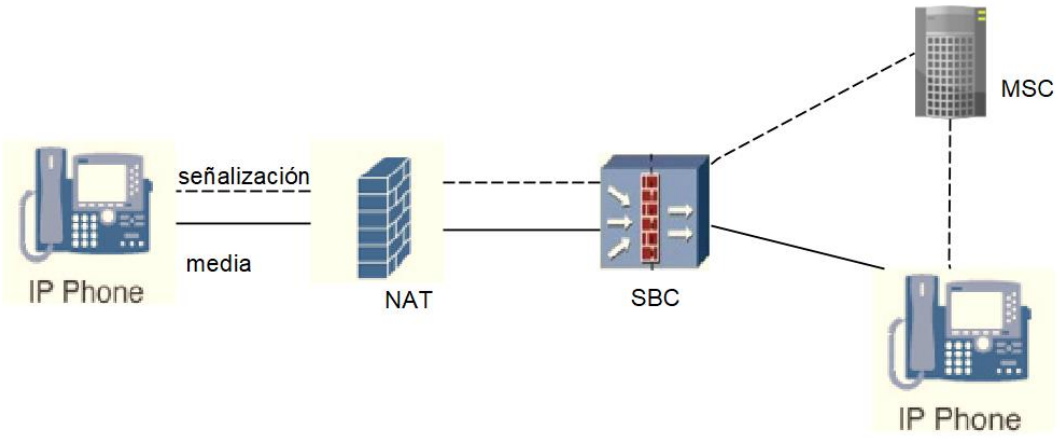
Ahora bien, se han definido los conceptos básicos de un SBC, y, tal como se comentó, dependiendo de dónde se ubique puede tener varios casos de aplicación, a continuación, se verán los escenarios de implementación de los SBC.

3.9. A-SBC – UNI – User-Network-Interface

En esta implementación justo en la frontera entre los usuarios (UE, *user equipment*) y la red del operador, el SBC reenvía los mensajes SIP desde los usuarios hacia el *core* del operador o viceversa.

Figura 17.

SBC implementado como un A-SBC



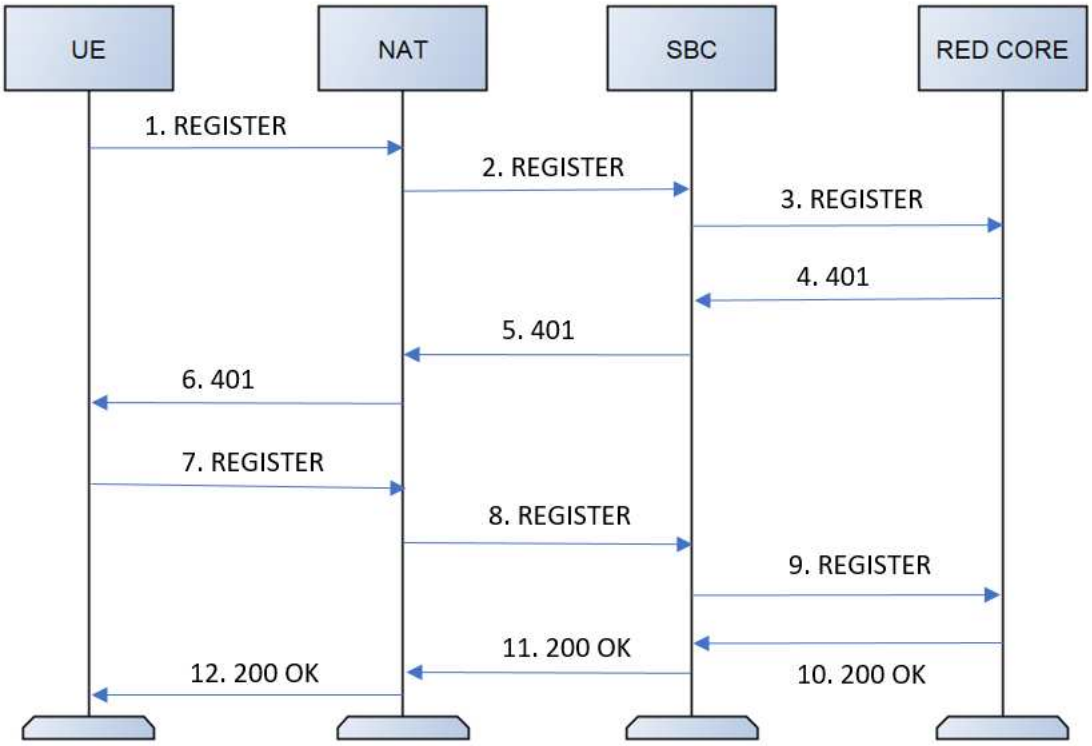
Nota. Implementación de SBC. Elaboración propia, realizado con yEd Graph Editor.

3.9.1. Registración SIP

La registración SIP corresponde a un proceso en el cual los dispositivos de los usuarios (UE) inician solicitudes de subscripción a la red a través del SBC para autorización.

Figura 18.

Registración SIP



Nota. Registración SIP. Elaboración propia, realizado con yEd Graph Editor.

- Un UE envía un mensaje de solicitud SIP *Register* hacia el SBC. En este mensaje se incluye la dirección de origen, puerto y dirección IP, puerto en el parámetro *contact* del encabezado. El mensaje *Register* incluye los siguientes parámetros.

    - *Request-URI:* es una línea en el encabezado que indica el nombre de destino para la solicitud *Register,* por ejemplo: sip:ejemplo.com.

- *To:* especifica el IMPU de un usuario registrado, por ejemplo: sip:+50255555555@ejemplo.com.

- *Contact:* especifica la dirección de contacto de un usuario registrado.

- *Via:* guarda el trayecto desde donde el mensaje *Register* es transmitido para así que la respuesta viaje por el mismo trayecto.

- El NAT tiene una IP pública y puerto, y cambia la dirección de origen en el mensaje *Register* en el encabezado por la dirección pública que el posee. Después este mensaje *Register* lo envía hacia el SBC.

- Después de recibir el mensaje *Register* el SBC tiene una dirección más puerto del lado *core,* es decir del lado del operador de servicio. Cambia la dirección que provenía del NAT y le coloca la propia del lado *core.* Entonces el SBC envía este mensaje hacia el *core* de la red del operador.

- El *core* de la red del operador responde con un SIP 401 (*unauthorized*) hacia el SBC, esto indica que requiere autenticación por parte del SBC.

- Después de recibir este SIP 401, el SBC vuelve a cambiar las direcciones IP ahora las del lado del NAT y envía el mensaje hacia el dispositivo NAT.

- Cuando el NAT recibe el mensaje 401, cambia la dirección IP con las privadas y envía el mensaje hacia el UE.

- Cuando el UE recibe el mensaje 401, autentica en el *core* de la red del operador de acuerdo con una llave local. El UE calcula la respuesta (RES)

de acuerdo con la llave de autenticación y crea un nuevo mensaje *Register* llevando consigo el RES, entonces el UE envía este *Register* de nuevo hacia el NAT en el mismo camino tomado anteriormente.

- Cuando el NAT recibe de nuevo el mensaje *Register* vuelve a cambiar la dirección IP privada por la que él tiene como pública, y envía el mensaje hacia el SBC.

- El SBC, como se vio anteriormente, cambia la IP NAT por la que tiene como dirección del c*ore* de la red del operador, y envía el mensaje hacia el *core.*

- El *core* de la red, como recibe ahora la llave de autenticación, genera la respuesta devolviendo un mensaje SIP 200OK.

- Al recibir el SBC el 200OK vuelve a cambiar las direcciones IP hacia las que conoce como direcciones del lado del NAT,

- El NAT recibe el 200OK y cambia la dirección por la privada el UE.

- El registro se completa exitosamente.

Existen un sinnúmero de dispositivos finales (UE) con diferentes capacidades de media, el SBC negocia la media en el proceso de establecimiento de una llamada SIP, para que se puedan comunicar distintos tipos de UE con el mismo tipo de media y códec. En este proceso de llamada SIP el SBC puede soportar los siguientes procesos:

- Bloqueo de media temprano: después de que se complete la negociación de solicitud/respuesta de SDP y antes de que el destinatario envíe un mensaje 200 OK, tanto el usuario que llama como el que es llamado sabrán sus direcciones de media. En este punto ambos pueden hablar entre ellos, como la llamada se empieza a tarificar solo después de recibir el 200OK por parte del usuario destino, existe una posibilidad de que usuarios maliciosos tomen ventaja de esta situación y realicen llamadas sin cobro. El bloqueo de media temprano aplica en el escenario del A-SBC. Con este proceso el SBC descarta los mensajes de media originados o recibidos por el destino antes de recibir el mensaje 200OK por parte del destino. Previniendo así que usuarios malintencionados realicen llamadas sin cobro.

- Temporizador de sesión: este es un mecanismo por medio del cual un usuario o un elemento de la red envía una solicitud de refresco de sesión para detectar si un elemento está aún activo en la red. Tanto el SBC como el *core* de la red pueden terminar la sesión cuando la comunicación o algún elemento falla, para prevenir que la sesión siga consumiendo recursos. En una red *core* la media se separa de la señalización, debido a esto es posible que la media esté inalcanzable pero la señalización no se libere, provocando con esto que la tarificación continúe. Con un temporizador de sesión, las sesiones son terminadas en tiempo evitando cobros y uso de recursos inadecuados.

- Detección de media: cuando el SBC detecta que no existe media envía un mensaje SIP BYE al UE y a la parte del *core.*

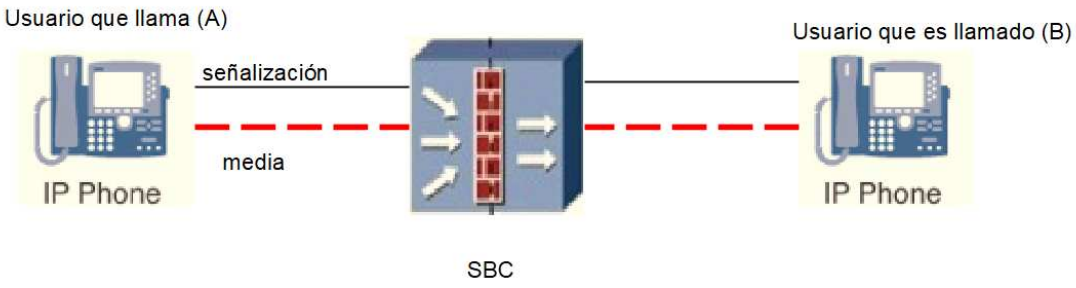
- Servicios suplementarios: los SBC pueden manejar los servicios suplementarios, como por ejemplo llamada en espera, traslado de llamada, llamada tripartita, conferencia y recepción/envío de mensajería corta (SMS).

3.9.2. Políticas de media

Estas políticas permiten al SBC a controlar las capacidades de media, esto permite el mejor aprovechamiento de los recursos de la red.

Figura 19.

Políticas de media



Nota. Políticas de media. Elaboración propia, realizado con yEd Graph Editor.

Entre estas políticas están las siguientes:

- Bloqueo de media temprana: como se ha visto antes de que el usuario que es llamado responda con un mensaje SIP 200OK para completar el flujo de la llamada, el SBC no reenvía los paquetes de media hacia o desde el usuario que es llamado, esto para que usuarios mal intencionados no generen llamadas no tarificadas. Cabe resaltar que el SBC no bloquea los paquetes de media hacia o desde el número que llama, esto para que el

usuario que llama tenga el *ring back tone,* el cual utiliza estos paquetes de media. Solo se bloquean paquetes de UDP mas no así los transmitidos por TCP.

- Verificación del tipo y ancho de banda en la media: el SBC restringe los tipos de paquetes de media, como por ejemplo paquetes de video. El ancho de banda también es restringido para cada paquete de media, esto previene que se sobrepase el ancho de banda de la media.

- Verificación de códec: el SBC restringe los *codecs* de audio y videos que son permitidos en la red. También asegura el uso de *codecs* con prioridad para que sean utilizados en las llamadas.

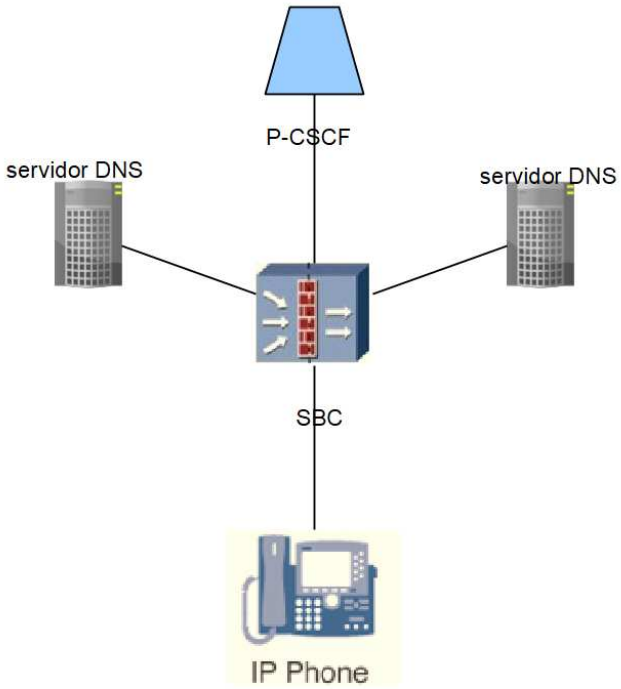
- Detección de media: tal como se ha visto en un escenario A-SBC, cuando no existen paquetes de media o se detecte una media anormal, el SBC envía un mensaje SIP *bye* hacia ambos extremos, el UE y el *core* de la red. Con lo anterior se libera la llamada y se asegura la buena tarificación.

3.9.3. Consultas de DNS

El sistema de nombre de dominio es utilizado para enrutar mensajes de señalización haciendo la traducción entre nombres de dominio y direcciones IP. En un escenario de A-SBC después de recibir el mensaje SIP de solicitud *Register*, el SBC consulta los servidores DNS sobre la dirección del P-CSCF, parte de la solución de un IMS, entonces al recibir la dirección envía el *Register* hacia el P-CSCF.

Figura 20.

Consultas de DNS



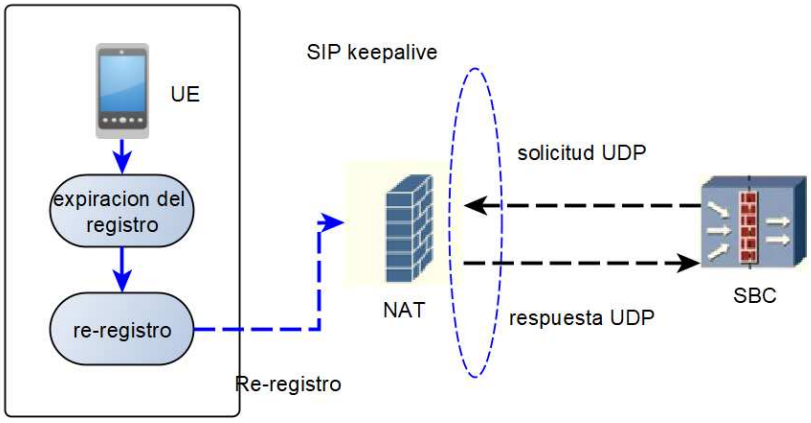
Nota. Consultas de DNS. Elaboración propia, realizado con yEd Graph Editor.

3.9.4. SIP keepalive

El tiempo de volverse a registrar o como se conoce re-registro de un UE es más grande que el tiempo en que NAT guarda la información. El SBC, al ser un dispositivo heurístico, aprende las direcciones que origina el UE después del NAT. El NAT no podrá mantener la información del UE si el periodo de registración es muy alto, y los paquetes se perderán, para prevenir esto el SBC periódicamente envía paquetes UDP hacia el NAT para refrescar el tiempo en que se mantiene la información del UE.

Figura 21.

Diagrama de keepalive

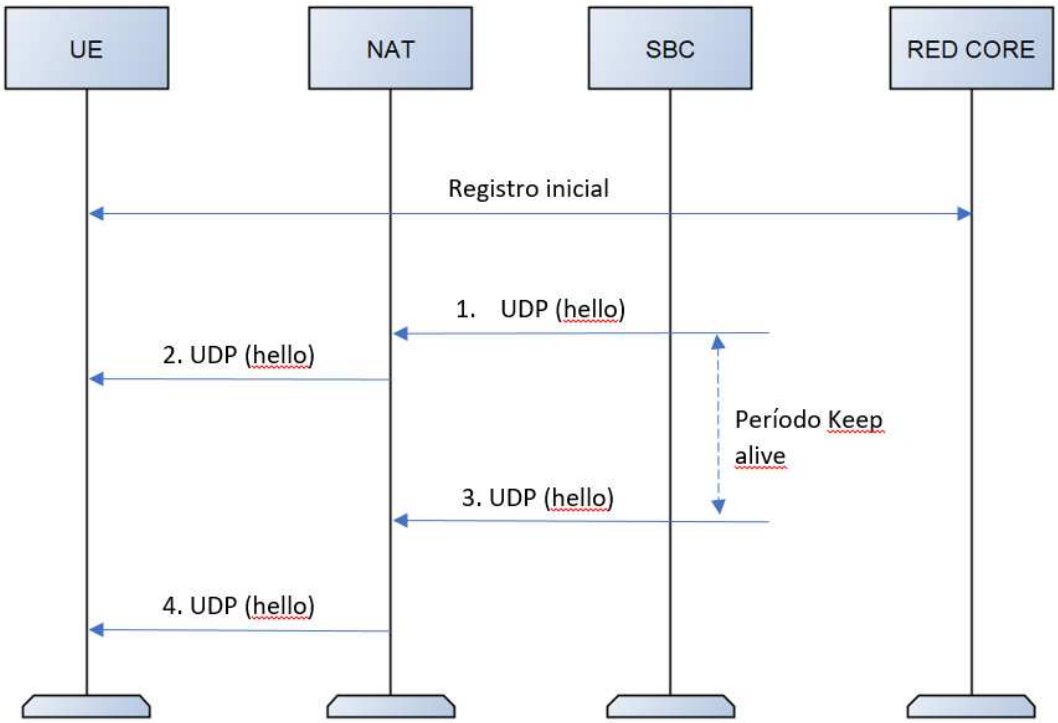


Nota. Diagrama de *keepalive.* Elaboración propia, realizado con yEd Graph Editor.

Los paquetes UDP pueden ser del tipo *hello packet* (definido por el usuario), SIP re- *Register*, paquete STUN y paquete CLRF utilizado en TCP.

Figura 22.

Diagrama de flujo de keepalive



Nota. Diagrama de flujo de *keepalive.* Elaboración propia, realizado con yEd Graph Editor.

- Después de que un usuario se registra exitosamente, el SBC envía paquetes UDP hacia la dirección pública alojada en el NAT.

- NAT retransmite estos mensajes hacia el UE y restablece el tiempo.

3.9.5. NAT traversal

Tal como se ha visto, las empresas utilizan direcciones privadas para sus comunicaciones internas y utilizan pocas direcciones públicas para comunicarse

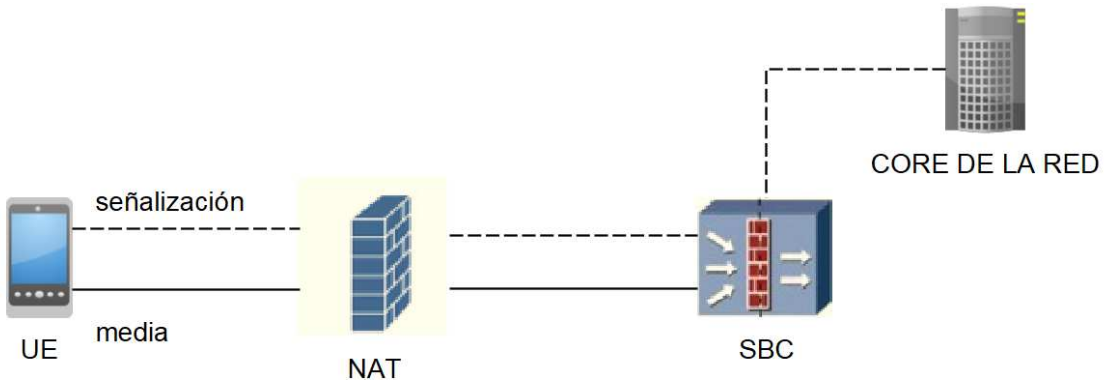
con las redes externas, ya que un paquete con dirección privada no puede enrutarse a través de Internet, NAT traduce la dirección privada a una dirección pública.

NAT ayuda a procesar y filtrar paquetes en las capas de red y transporte, cuando un paquete pasa a través de NAT, las direcciones de las capas de red y transporte cambian, pero las direcciones de la capa de aplicación se mantienen privadas. Si se retorna un paquete a la capa de aplicación el paquete fallará en alcanzar su destino.

La dirección SIP se encuentra en la capa de aplicación, para asegurar la transmisión de paquetes el SBC implementa NAT traversal cambiando las direcciones y puertos recibidos desde el UE.

Figura 23.

NAT traversal



Nota. NAT traversal. Elaboración propia, realizado con yEd Graph Editor.

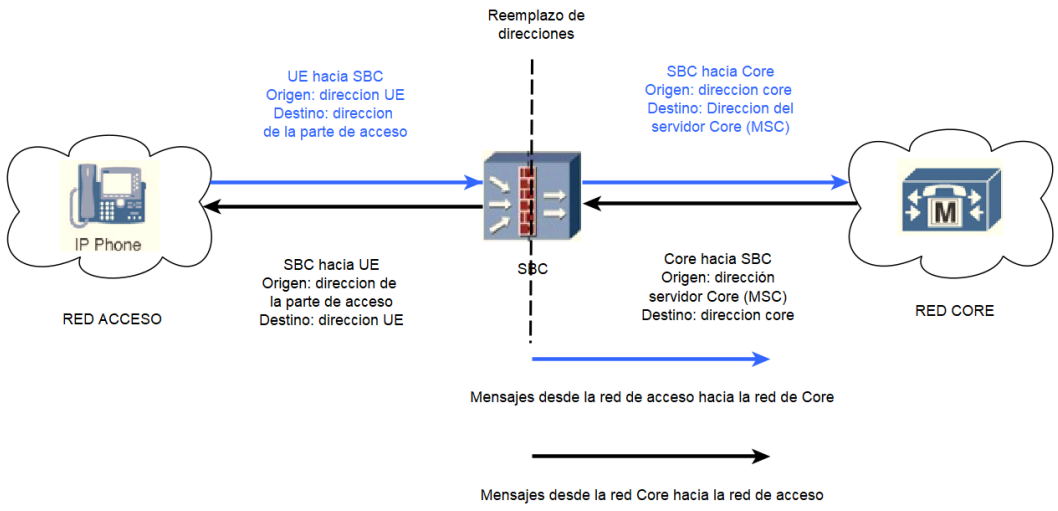
3.9.6. Ocultación de topología

Durante las llamadas SIP que se realizan el *core* de la red puede quedar expuesto, los atacantes pueden estar sondeando la red para así poder acceder de forma anómala al *core.*

En esta configuración, donde el SBC se despliega entre la red pública y la red privada, el SBC oculta el *core* de la red de la parte del acceso y viceversa. Como se puede observar, en esta configuración se previene que los usuarios puedan tener acceso al *core* de la red. Brinda una arquitectura de protección a la red. Los SBC implementan esta ocultación de topología haciendo una traducción entre las direcciones de acceso y de *core,* durante el procesamiento de los mensajes SIP.

Figura 24.

Ocultación de topología



Nota. Ocultación de topología. Elaboración propia, realizado con yEd Graph Editor.

En la red del *core* el SBC implementa la ocultación de la topología haciendo uso de NAT para traducir las direcciones IP y puerto para que el UE solo pueda tener acceso a las direcciones del SBC

En la red de acceso el SBC implementa la ocultación de la topología cambiando las direcciones IP y puerto en la capa de transporte. Y las direcciones IP y los encabezados SIP en los mensajes SIP, para que así el *core* de la red no tenga acceso a las direcciones y puertos del UE.

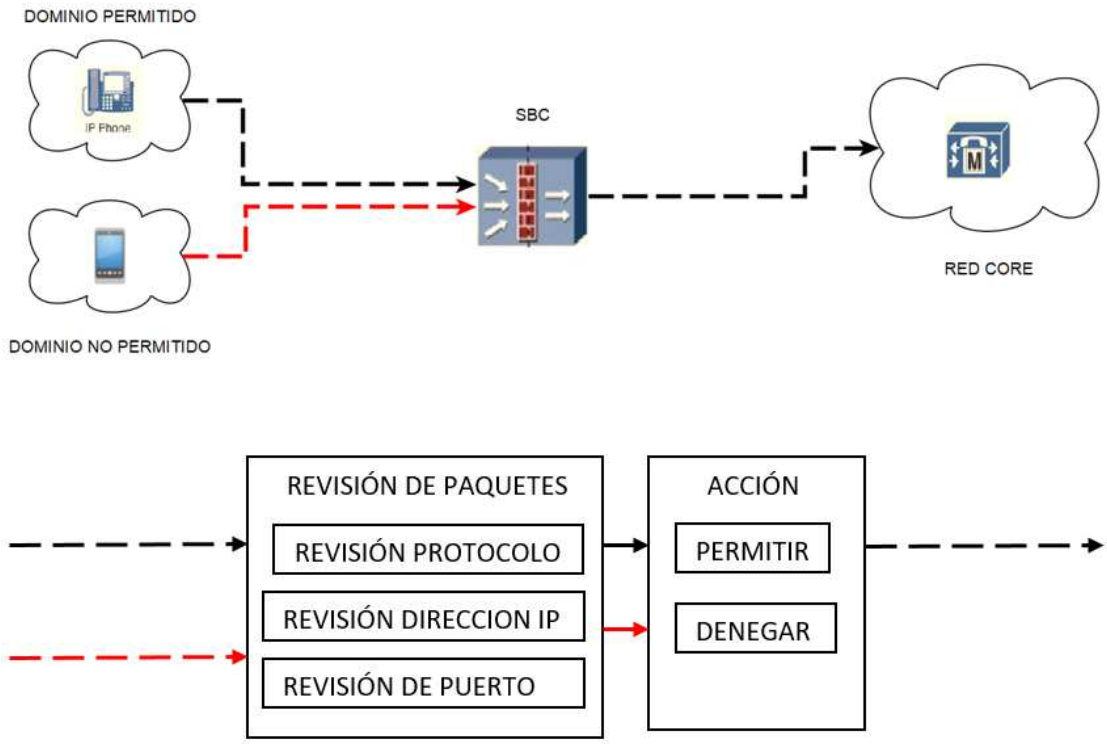
3.9.7. Seguridad en la capa IP

Filtrado de paquetes basado en listas de control de acceso.

Las listas de control de acceso definen reglas que son usadas para filtrar paquetes IP. El SBC hace una revisión de los paquetes y, si no cumplen las reglas, los descarta.

Figura 25.

Seguridad en la capa IP



Nota. Seguridad en la capa IP. Elaboración propia, realizado con yEd Graph Editor.

3.9.8. Defensa contra ataque DoS o DDoS

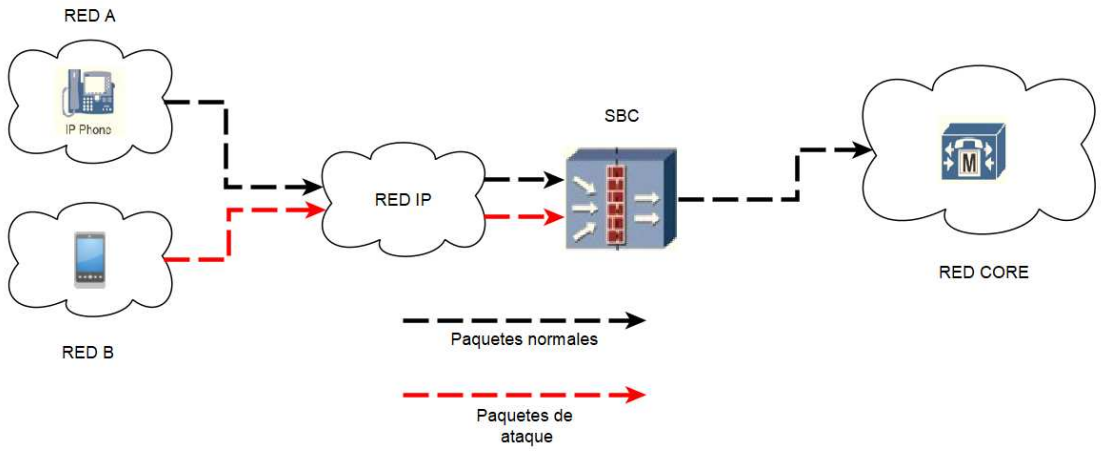
Todos los mensajes de señalización y media son transportados en paquetes IP. Un atacante puede lanzar un ataque en la capa de IP enviando un gran número de paquetes IP hacia un destino para tratar de agotar sus recursos. También puede enviar paquetes malformados para causar un fallo al servidor, al tratar de procesar estos paquetes. El SBC tiene la capacidad de detectar, clasificar y filtrar estos paquetes y que no resulten en un riesgo para la red. Una vez identificados los paquetes, el SBC genera un mecanismo de defensa, como

colocar en una lista negra la IP más el puerto del origen, al exceder el umbral de eventos permitidos.

Los ataques DoS son enviados por un solo usuario, mientras que los DDoS son enviados por múltiples atacantes. En la mayoría de los casos los atacantes envían estos ataques enviando muchos paquetes hacia un puerto conocido, en SIP el puerto 5060 tratando de agotar los recursos y que el sistema falle procesando los paquetes válidos.

Figura 26.

Defensa contra ataque DoS o DDoS



Nota. Defensa contra ataques. Elaboración propia, realizado con yEd Graph Editor.

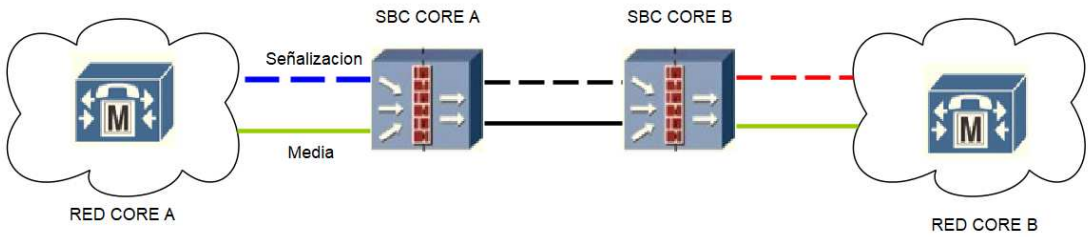
3.10. I-SBC – Network-Network Interface (NNI) SBC

En esta configuración el SBC actúa como un punto de interconexión entre redes, actualmente la mayoría de los operadores se han migrado a arquitecturas basadas en SIP, sin embargo existen operadores que utilizan el protocolo SS7.

Para mitigar los costos y complejidad para interconectar las redes, el SBC se configura como un punto de interconexión. Por lo general, entre distintos operadores, cada uno tiene su SBC, los cuales se interconectan entre sí para lograr comunicarse.

Figura 27.

I-SBC – Network-Network Interface (NNI) SBC



Nota. Detalle del diagrama I-SBC. Elaboración propia, realizado con yEd Graph Editor.

En la imagen se puede apreciar que cuando la red *core* A genera una llamada la entrega hacia el SBC, este recibe la llamada y la envía hacia la red *core* B. En esta arquitectura el SBC actúa como servidor para el usuario A y como cliente para el *core* de red B.

3.10.1. Políticas de media

Las políticas de media permiten al SBC controlar las características de la media, tal como media temprana, tipos de media, *codecs* de media y ancho de banda.

Entre estas políticas están las siguientes.

- Revisión del tipo y ancho de banda de la media: el SBC verifica los tipos de media y restringe paquetes de media específicos. También restringe el ancho de banda para algún tipo de media y previene el uso excesivo de ancho de banda.

- Verificación de *codec:* el SBC restringe los *codecs* de audio y video que viajan a través de la red. También le da prioridad a los *codecs* que son más recomendables.

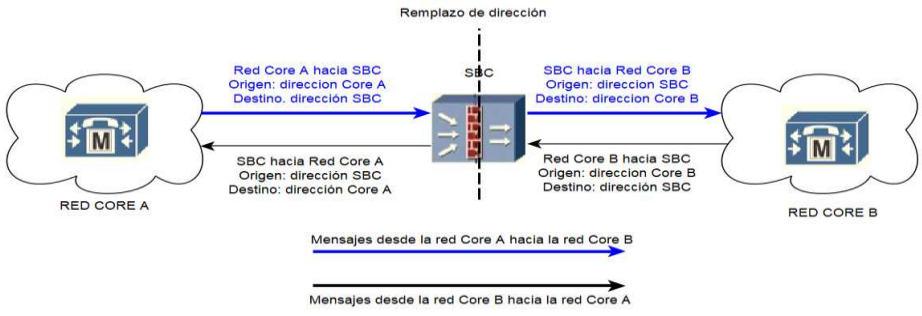
- Detección de media: cuando hay señalización, pero no media, el SBC lo detecta y envía un mensaje SIP BYE para terminar la sesión.

3.10.2. Ocultación de topología

En cuanto a este tema, es necesario recordar que en esta topología el SBC oculta la topología de cada red, una de la otra. Se implementa cambiando la IP de origen por la dirección del SBC.

Figura 28.

SBC oculta la topología de cada red



Nota. SBC oculta la topología. Elaboración propia, realizado con yEd Graph Editor.

3.10.3. Seguridad en la capa IP

También es muy importante los aspectos que conciernen a la seguridad en la capa IP, lo cual está relacionado tanto con todo lo explicado anteriormente como con los incisos que a continuación se desarrollan.

3.10.4. Filtrado de paquetes basado en listas de control de acceso

Como se vio en la configuración del A-SBC, en el I-SBC también se tiene el filtrado de paquetes basado en listas de control de acceso, en el cual se validan los paquetes por medio de reglas y se descartan aquellos que no las cumplan. Este proceso protege los puertos y servicios de ataques.

3.10.5. Defensa contra ataque DoS o DDoS

De la misma forma que en la parte de acceso, se cuenta con la defensa ante ataques DoS, teniendo umbrales en los cuales, al llegar al máximo, se colocan en listas negras las IPs, o dominios de aquellos que estén ocasionando los ataques.

El SBC detecta el tráfico proveniente de una troncal que no conoce y bloquea los paquetes provenientes de esta troncal que no tiene en su lista.

4. CAPÍTULO 4

4.1. Operación y mantenimiento

Dependiendo del operador, la gestión puede llevarse de una manera muy amigable con el usuario, hoy en día los proveedores se están orientando a la virtualización, y una de sus principales ventajas es una alta disponibilidad y un entorno web para la gestión. En años anteriores únicamente eran gestionados los equipos vía comando, lo que se conoce como CLI o *comand line interface.* Y los equipos aún cuentan con CLI pero ya en una versión más actualizada. La operación y mantenimiento de los SBC incorpora la gestión de fallas, gestión de configuración, gestión de desempeño y gestión de seguridad (Huawei Technologies, 2017).

4.1.1. Gestión de fallas

En este apartado se incluye la detección de la falla, la localización de la misma y su resolución. Una falla puede ser de *hardware, software* o de aplicación. En la gestión de las fallas el SBC cuenta con un automonitoreo del sistema, esto significa que el SBC periódicamente verifica el uso de sus recursos, procesos, memoria, espacio en disco, entre otros, también detecta y reporta cualquier evento o falla en el dispositivo, hay proveedores que han incluido sonidos a las alarmas para una mejor apreciación, también ayuda para poder trabajar la alarma que se presenta.

Trazados: una de las principales herramientas para solucionar problemas de servicio son los trazados. Algunos proveedores ya traen incorporado en sus

sistemas servidores dedicados para trazados, los cuales se pueden visualizar en tiempo real, sin embargo, también pueden descargarse y analizarse por aparte con otras herramientas.

4.1.2. Gestión de configuración

Los SBC proporcionan un sistema de configuración basado en comandos, estos comandos permiten a los operadores el monitoreo y gestión del equipo. Los comandos permiten crear, borrar, modificar, entre otras actividades. Existen sistemas que permiten ejecutar comandos por lotes, para que sea más fácil la ejecución de configuraciones masivas.

4.1.3. Gestión de desempeño

Una de las herramientas más útiles para validar si en la red existe algún problema, o validar desde qué fecha se viene presentando el inconveniente, son los principales indicadores de desempeño o KPI por sus siglas en inglés. Estos indicadores pueden dar una representación de la calidad del servicio al usuario, degradación de servicio o cantidad de baja de usuarios, y así tomar decisiones correctivas para mejorar la red.

Los SBC pueden traer integrado un servidor de KPI o su gestor como tal puede incluir funcionalidades para su análisis. De cualquier forma, el equipo recopila los KPI, los procesa y los despliega para tomar decisiones. Hay algunos conceptos que se deben tener en cuenta:

- Objeto de medición: son varios elementos físicos o lógicos para ser medidos.

- Ítem de medición: es la unidad básica de medición de desempeño.
- Periodo de medición: es el intervalo en que la tarea de medición va a generar la salida de los resultados. Pueden estar configurados en minutos, horas o incluso días.

4.1.4. Gestión de seguridad

Como se puede ver, el SBC en cualquiera de sus configuraciones es un elemento importante en la red, es por lo que se incluyen funcionalidades de seguridad del nodo en sí.

- Gestión de derechos: trata sobre los niveles de habilitación de los comandos para los operadores, puede haber usuarios solo con permisos para ver, para modificar o usuarios para ejecutar cualquier acción.

- Gestión de *logs:* el SBC puede contar con esta funcionalidad para realizar una auditoría de los comandos ejecutados por los operadores. Ayuda a la identificación de intrusos.

- Gestión de acceso: proporciona vías para poder acceder al equipo, conexiones ssh, *telnet,* servicio de sftp.

4.2. Rutinas de mantenimiento

Durante la operación de la red es importante realizar tareas periódicas en las que se incluyan revisiones al sistema para constatar que esté funcionando en óptimas condiciones y así poder prevenir cualquier falla.

4.2.1. Revisión del estado del equipo

También se conoce como *health-check,* es una listade tareas que se revisan en el equipo, estas pueden ser manuales o automáticas. Actualmente los sistemas ya traen consigo apartados automatizados para estas revisiones, la acción del operador será entonces revisar los resultados y con base en estos tomar acciones correctivas sobre el nodo. Algunas de las revisiones que se realizan son:

- Monitoreo del uso de CPU: independientemente si el sistema cuenta con interfaz gráfica o no, se debe validar la carga de los CPU, quizás con el comando TOP si es servidor Linux o visualmente en gráficos si el sistema lo permite.

- Monitoreo del uso de memoria: otra de las revisiones que se deben realizar es el uso de la memoria (*free* en Linux).

- Monitoreo del uso de disco duro: en ocasiones esta revisión es dejada de lado y ocasiona que los servidores se queden sin espacio para procesar las solicitudes, por eso la importancia de incluir esta verificación y como acción depurar directorios para liberar espacio.

- Revisión de los servicios: el monitoreo de los servicios se debe realizar periódicamente, con esto se puede validar que los servicios estarán disponibles para los usuarios.

4.2.2. Copia de respaldo (*backup*)

Es muy importante que ante cualquier eventualidad se tenga una copia de respaldo de toda la configuración del equipo, para que así, al existir un fallo que requiere cargar de nuevo toda la configuración, se tenga disponible. Usualmente estas copias se ejecutan automáticamente por el sistema a una hora definida por el sistema mismo o por el operador, en horario de bajo tráfico, pero también es importante realizarlo manualmente, ya que antes de una intervención del equipo se debe resguardar su configuración previa. Estos *backups* son resguardados en el propio equipo, pero lo más recomendable es que se tengan en medios externos, actualmente la mayoría de proveedores tiene habilitado el sftp para enviar los archivos a un medio de resguardo externo.

4.3. WireShark

Una de las principales herramientas para identificar un problema en la red es realizar un análisis de la propia red, es decir ir paso a paso a través de cada equipo que involucra el camino de una llamada para validar en cuál de ellos se está generando el fallo, a esto se le llama trazado, que es prácticamente realizar una captura en los equipos de todos los paquetes de señalización o media que están involucrados en la llamada (WireShark, s.f.).

Esta captura puede realizarse de varias formas, por los números de teléfono, por IP, sobre una troncal, entre otras. Va a depender de qué tan desarrollado esté el equipo en este aspecto.

Si hay un inconveniente de llamadas entre dos dispositivos, es posible llamar al dispositivo que origina la llamada “número de A” y al que recibe la llamada número de B, y se coloca el trazado dependiendo del inconveniente.

Muchos de los equipos SBC tienen la facilidad de colocar trazados sobre cualquiera de los dos números. Dependiendo del fabricante es posible encontrar que únicamente se pueden visualizar dentro del propio equipo, esto genera una dificultad, ya que, si en dado caso es necesario trasladar esta captura a otro personal y ellos no cuentan con el mismo programa no podrán visualizar el trazado, es por eso que conviene convertirlo a un formato que pueda ser leído por cualquier usuario.

Uno de los *softwares* que tiene disponibilidad es WireShark.

Alrededor de 1990 Gerald Combs se encontraba trabajando para un pequeño proveedor de servicios y en esa época los analizadores de protocolos eran de paga y no eran compatibles con todas las plataformas, para aliviar la carga de trabajo en el análisis y solución de problemas Combs se dispuso a crear un programa que pudiera visualizar lo que pasaba en la red y en 1998 lanzó la primera versión de lo que hoy se llama WireShark (WireShark, s.f.).

Es un programa OpenSource disponible para plataformas Windows y Unix. WireShark es un analizador de protocolos, analiza el tráfico en la red y para resolución de problemas es una excelente herramienta. Este programa implementa varios filtros los cuales ayudan a la búsqueda de los más de 1000 protocolos soportados en la actualidad. Es de una interfaz sencilla e intuitiva, en él se puede visualizar las capas de cada mensaje de un trazado. Dado lo anterior, es un programa muy versátil, que puede ser instalado por cualquier persona y ya que acepta varios formatos de trazados es posible compartir la información sin problema (WireShark, s.f.).

Uno de los formatos más utilizados es PCAP, el cual es un formato de archivo por paquetes, estos archivos permiten a los usuarios analizar el tráfico de

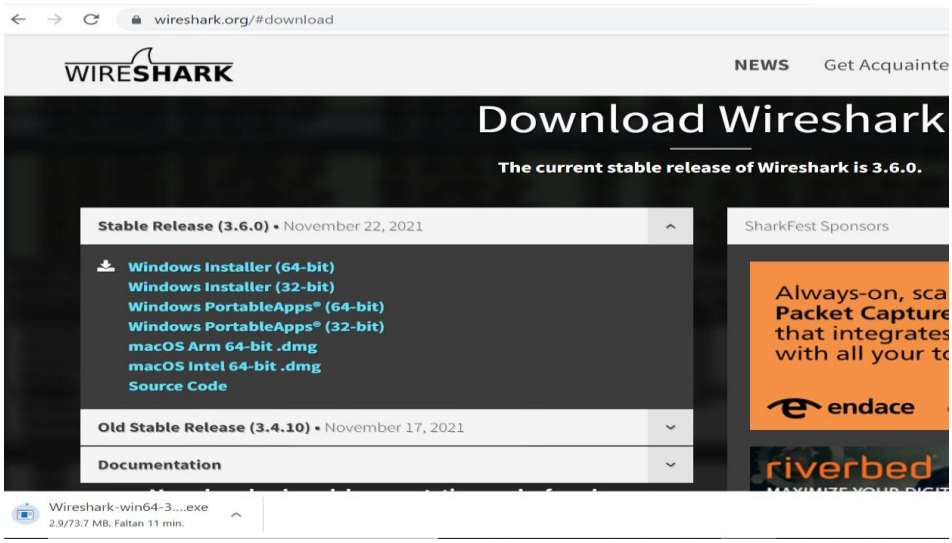
red para un trazado determinado, en modo fuera de línea, quiere decir que se coloca el trazado, se mantiene el tiempo necesario, se detiene y después se analiza.

4.3.1. Instalando WireShark

Dado que es un programa *OpenSource* se puede descargar gratuitamente en la página oficial mediante el enlace https://www.wireshark.org/#download. Dependiendo del sistema operativo se selecciona el archivo.

Figura 29.

Descargando WireShark

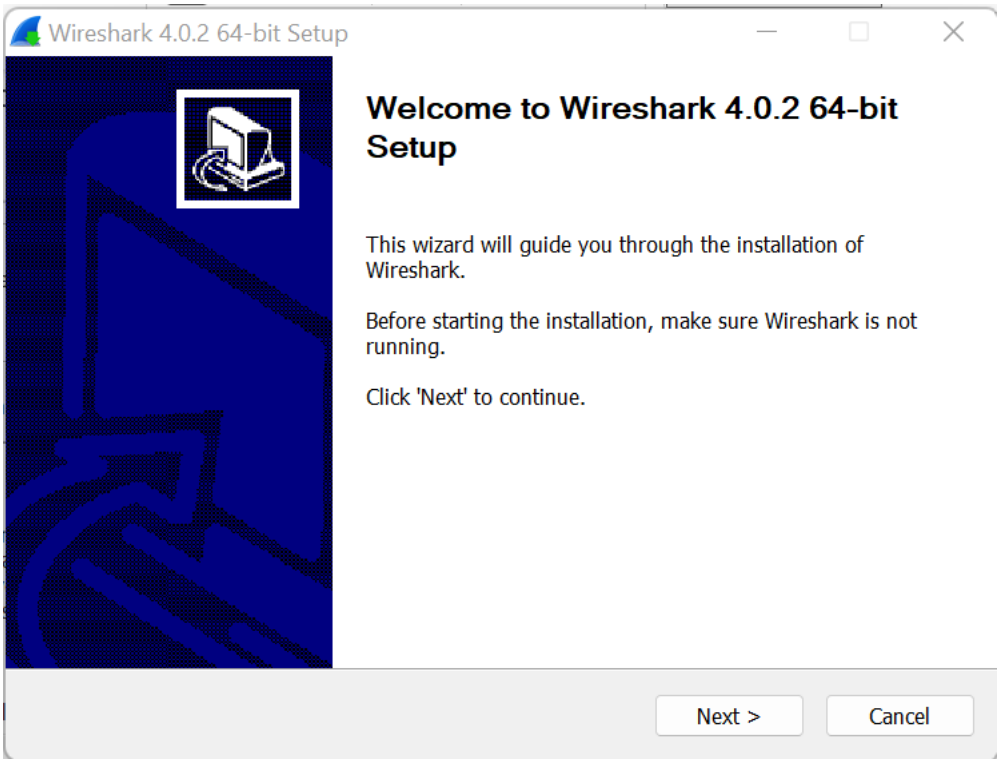


Nota. Descargando WireShark. Elaboración propia, realizado con WireShark.

Una vez teniendo el instalador en la máquina se procede a la instalación, es un *setup* amigable donde para una configuración básica solo se tiene que dar en siguiente hasta que el programa de instalación haya finalizado.

Figura 30.

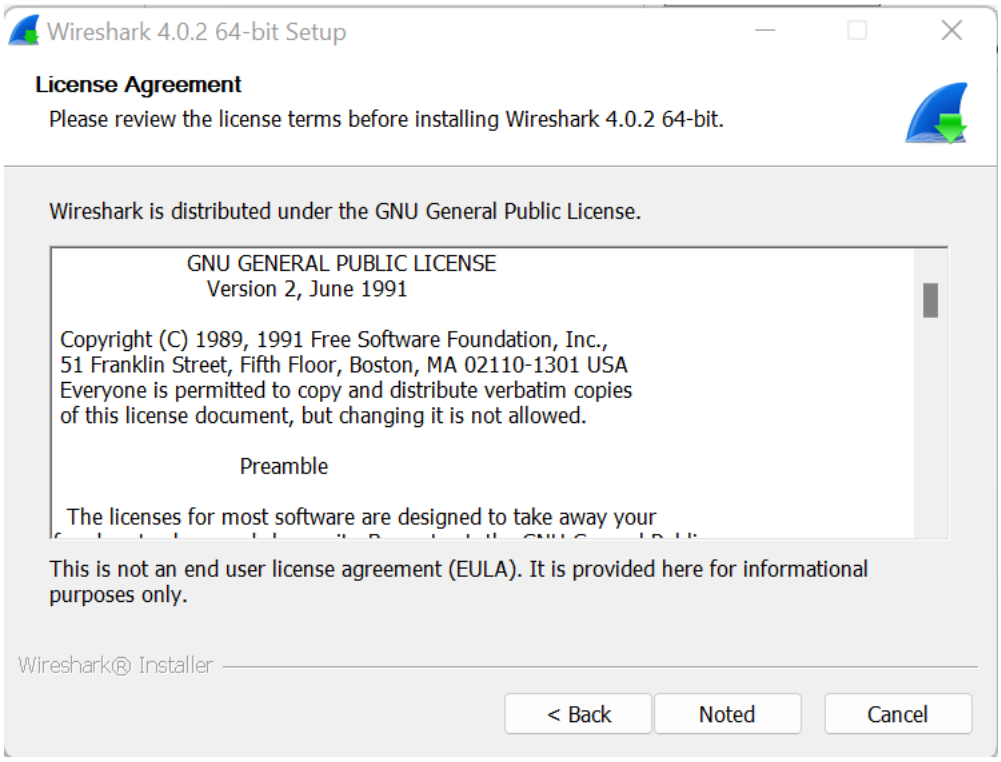
Instalando Wireshark



Nota. Instalando WireShark. Elaboración propia, realizado con WireShark.

Figura 31.

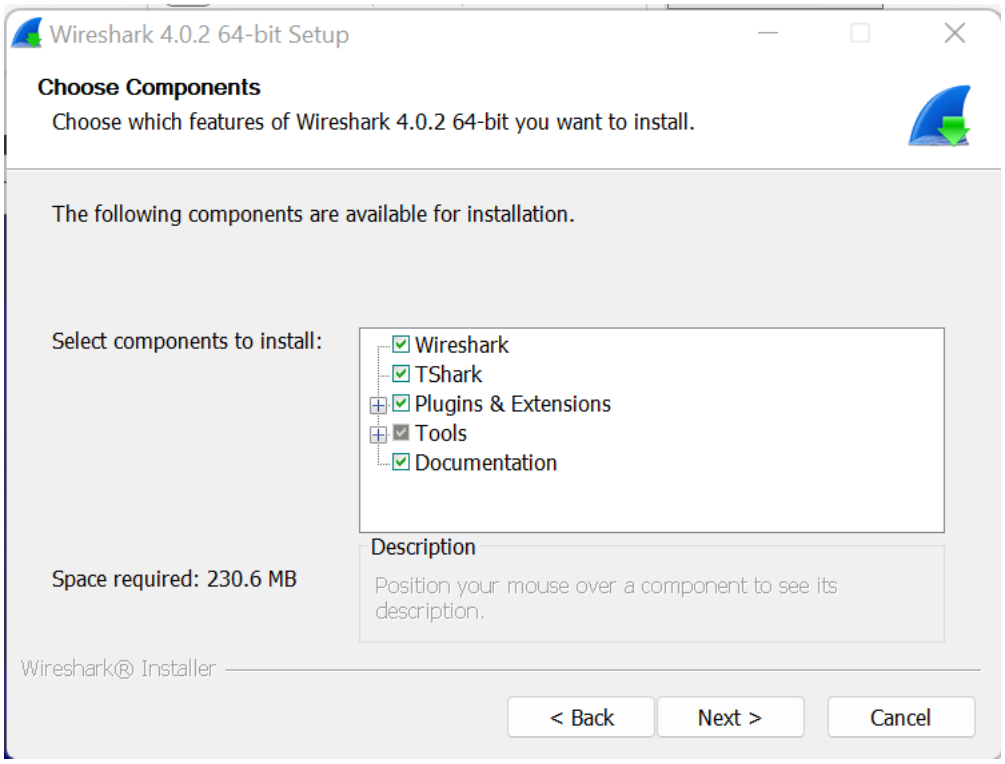
Términos de licencia



Nota. Términos de licencia. Elaboración propia, realizado con WireShark.

Figura 32.

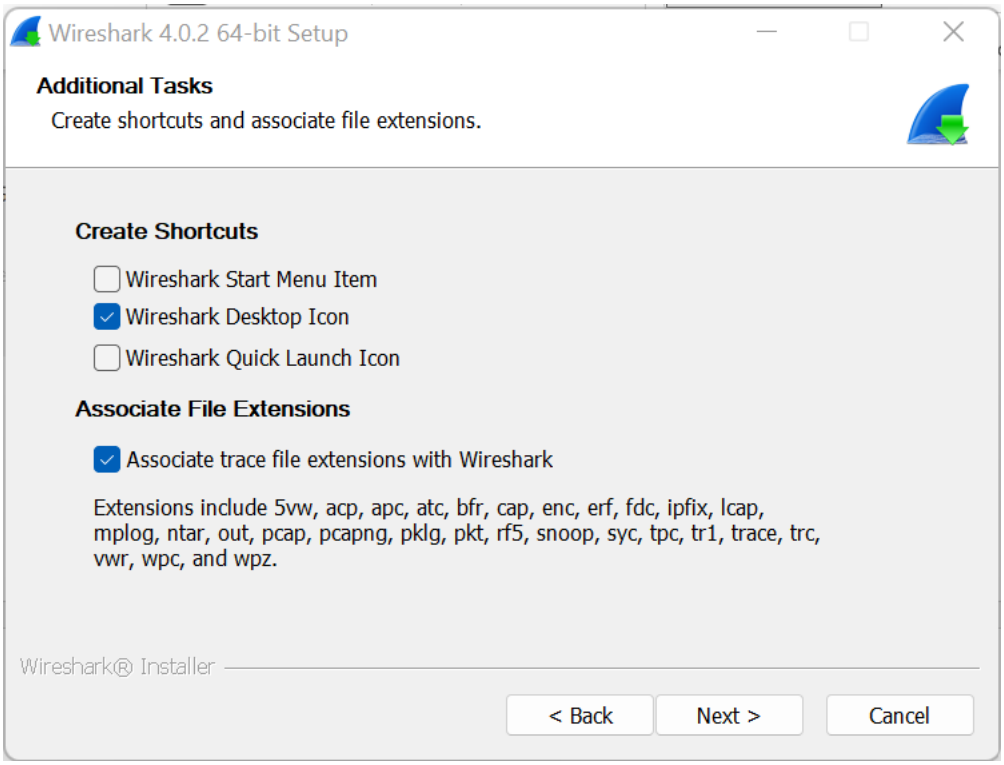
Seleccionar los componentes



Nota. Seleccionar los componentes. Elaboración propia, realizado con WireShark.

Figura 33.

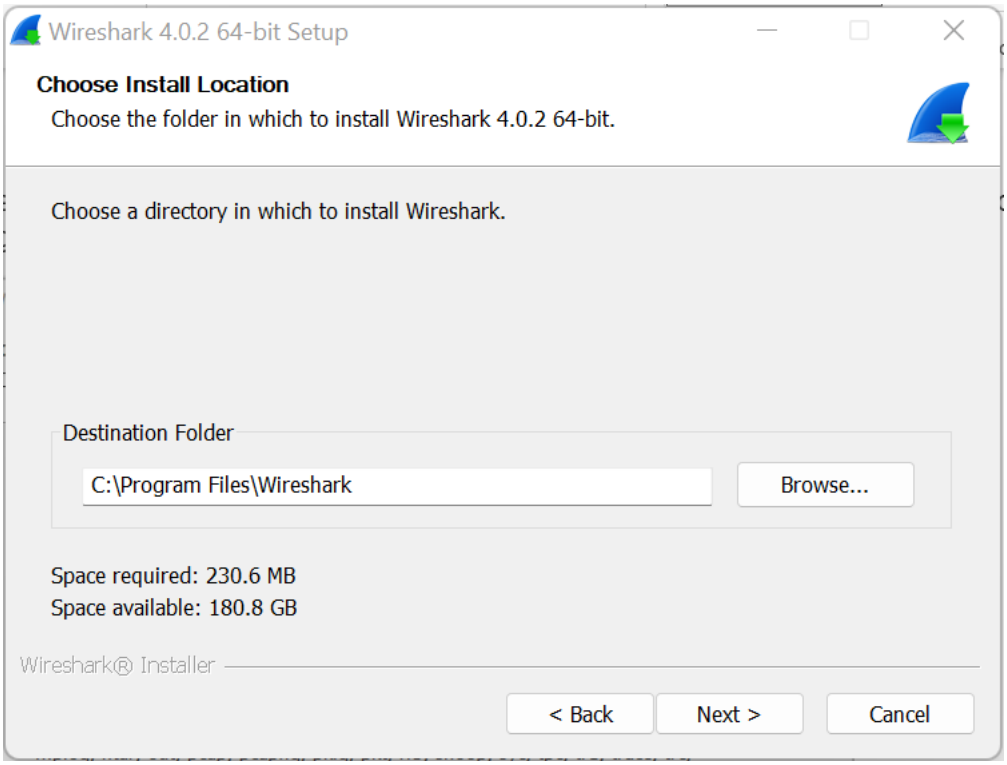
Asociar extensiones de archivos



Nota. Asociar extensiones de archivos. Elaboración propia, realizado con WireShark.

Figura 34.

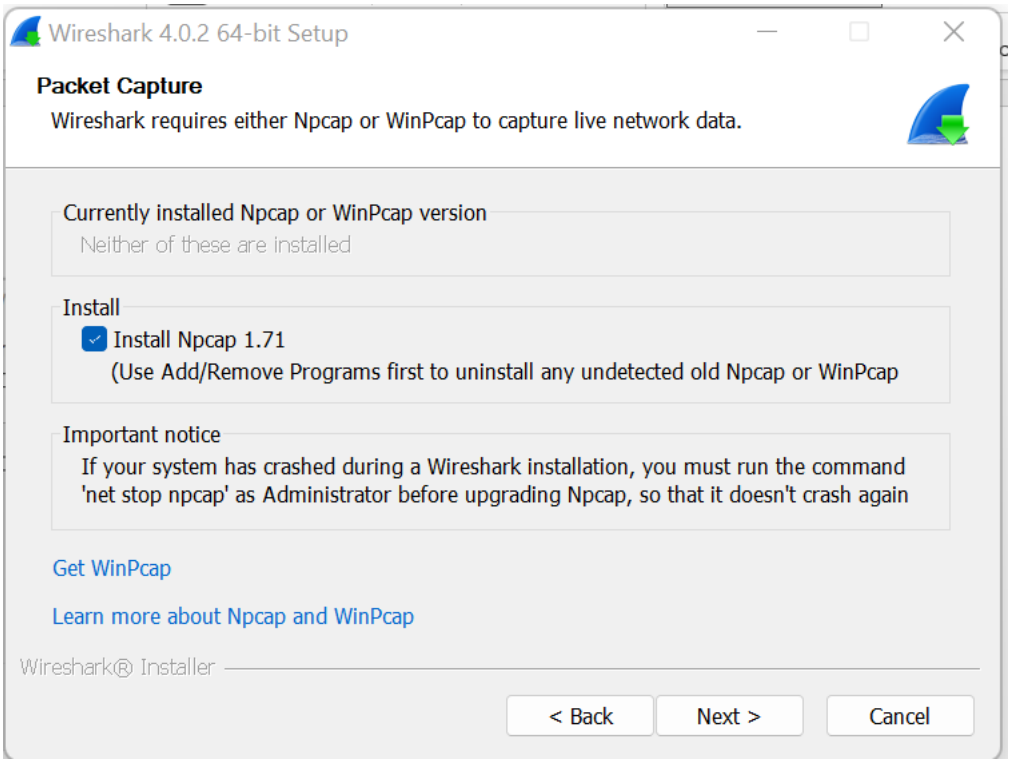
Localización de la instalación



Nota. Localización de la instalación. Elaboración propia, realizado con WireShark.

Figura 35.

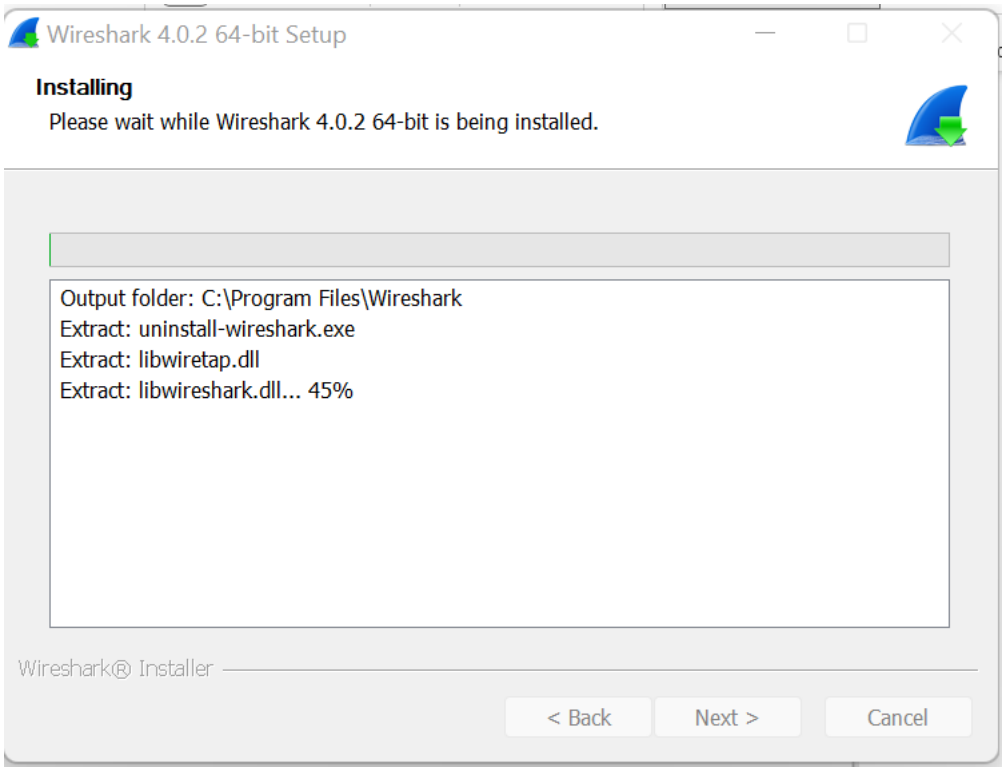
Opciones de captura en tiempo real



Nota. Opciones de captura en tiempo real. Elaboración propia, realizado con WireShark.

Figura 36.

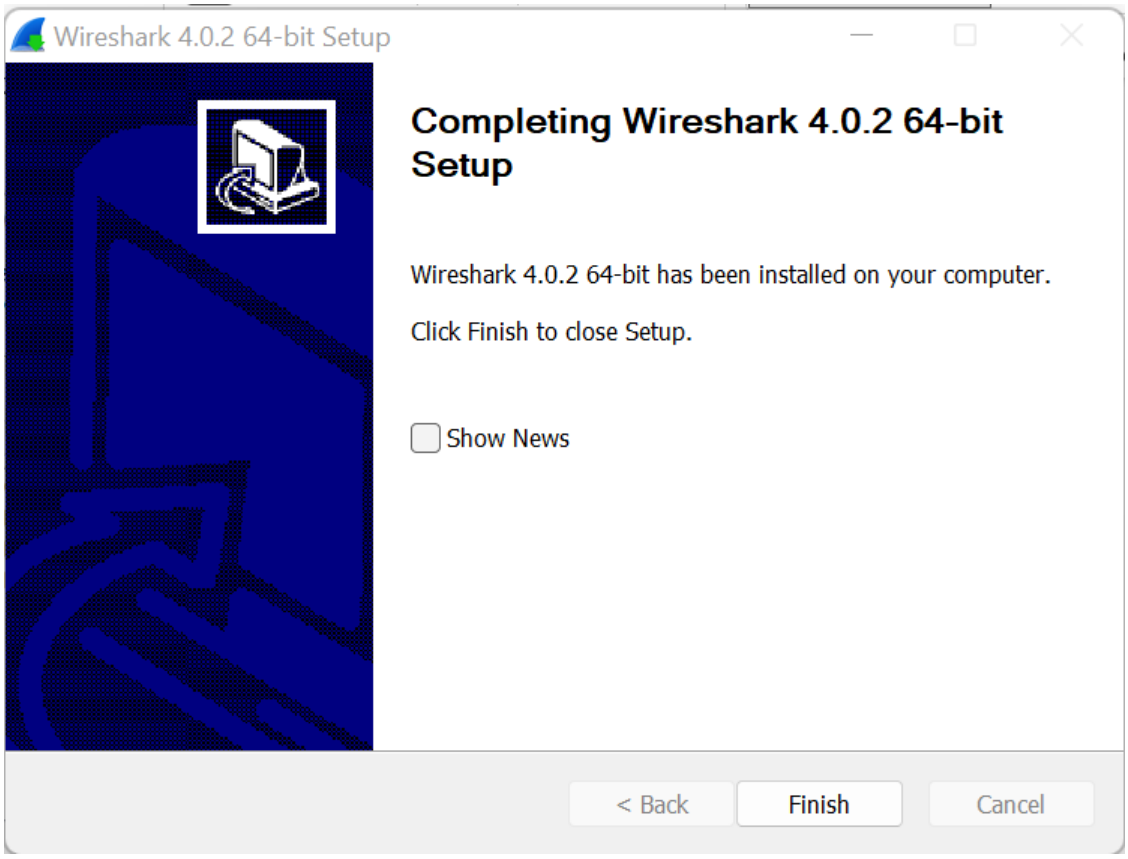
Proceder con la instalación



Nota. Proceder con la instalación. Elaboración propia, realizado con WireShark.

Figura 37.

Completando la instalación



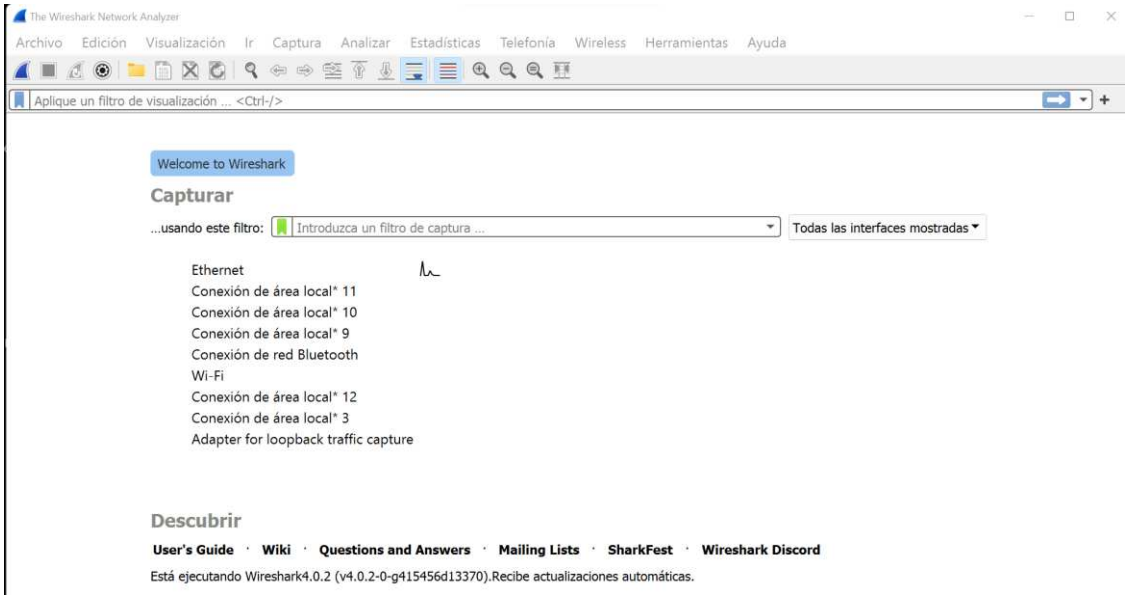
Nota. Completando la instalación. Elaboración propia, realizado con WireShark.

4.3.2. Configuración de WireShark

Como se ha visto, el software es muy utilizado y contiene muchas formas de buscar dependiendo del protocolo, sin embargo, se debe realizar algunas configuraciones antes de poder ver los mensajes que interesan en un trazado (Instituto Nacional de Ciberseguridad, s.f.).

Figura 38.

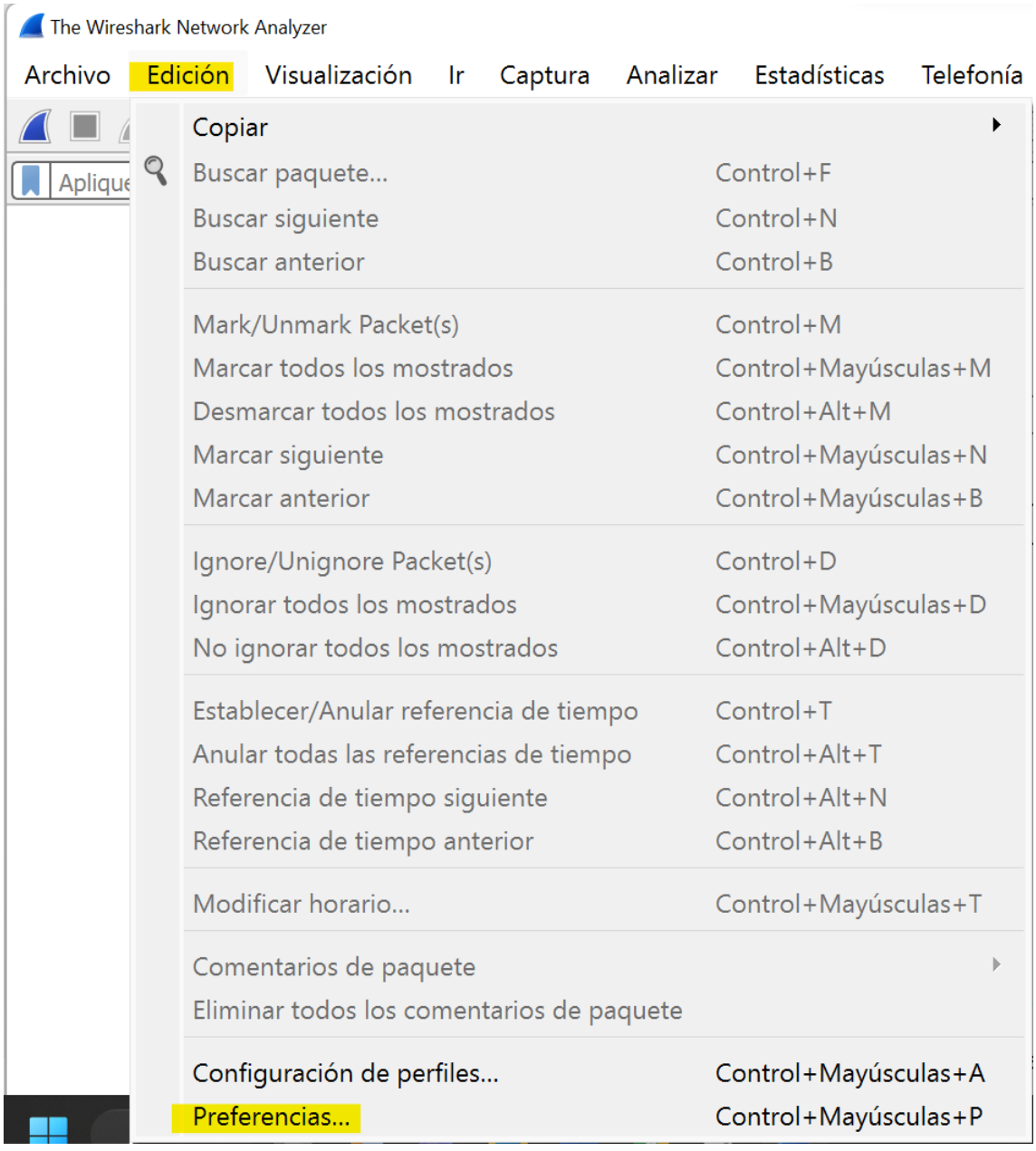
Inicio del programa



Nota. Inicio del programa. Elaboración propia, realizado con WireShark.

Figura 39.

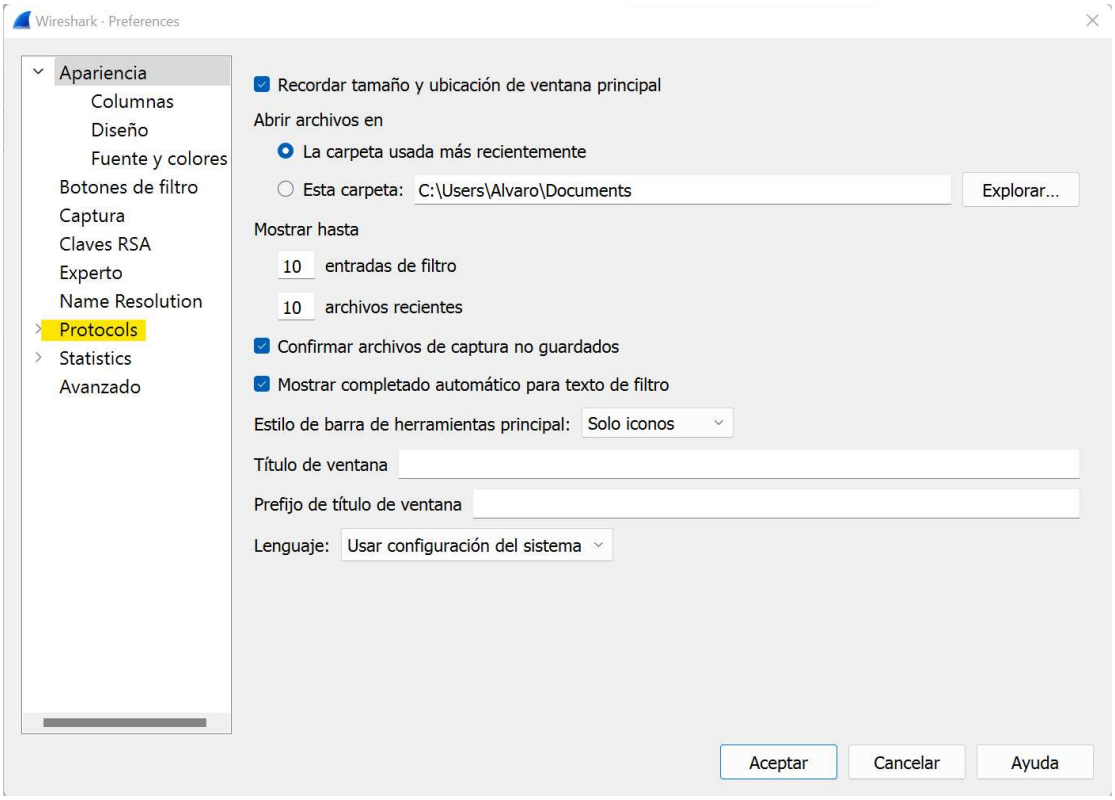
Seleccionar preferencias en menú



Nota. Seleccionar preferencias en menú. Elaboración propia, realizado con WireShark.

Figura 40.

Seleccionar protocols y expandir

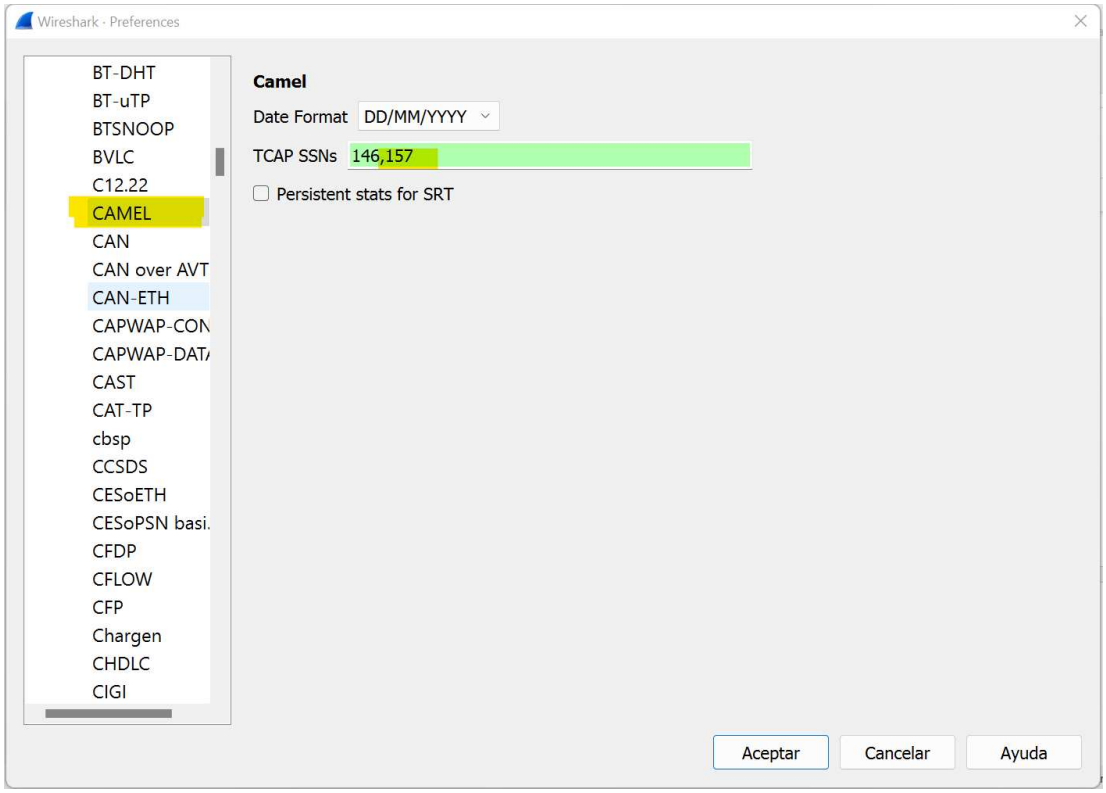


Nota. Seleccionar *protocols* y expandir. Elaboración propia, realizado con WireShark.

En la lista de los protocolos, se busca el protocolo CAMEL y se selecciona, aparecerá una imagen como la siguiente, en la cual después del número 146 se coloca sin espacios *,157.*

Figura 41.

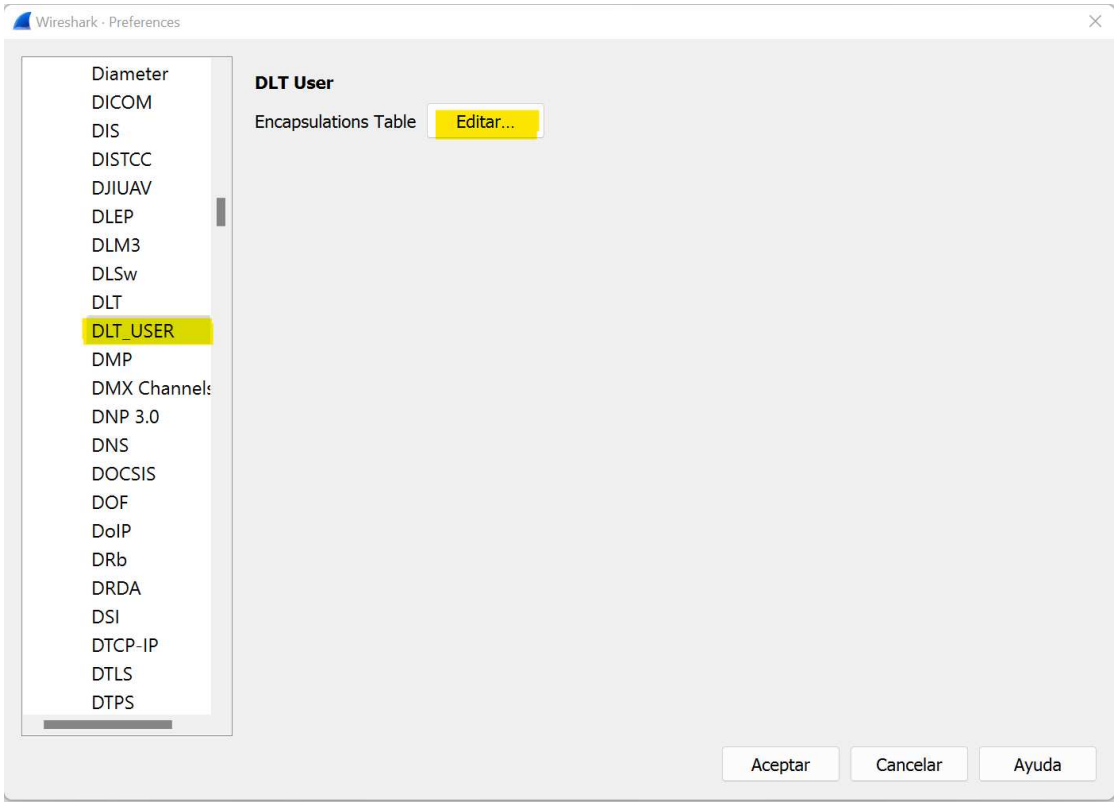
Buscar y seleccionar el protocolo CAMEL



Nota. Sobre el protocolo CAMEL. Elaboración propia, realizado con WireShark.

Figura 42.

Buscar y seleccionar DLT_USER

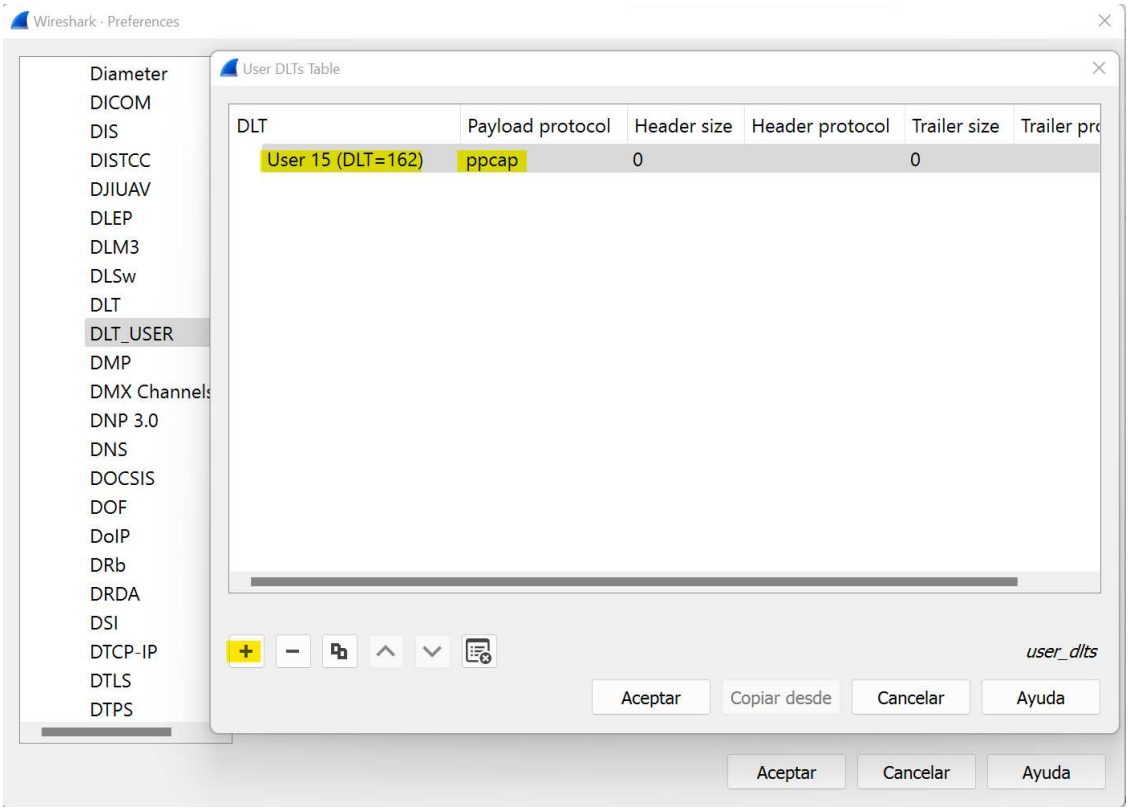


Nota. Buscar y seleccionar DLT_USER. Elaboración propia, realizado con WireShark.

En el menú que aparecerá es necesario dar click en el símbolo de más y en la primera columna DLT y buscar de la lista que presenta la opción *User* 15 (DLT=162). En la segunda columna llamada *Payload Protocol* agregar ppcap. Para finalizar se debe aceptar todas las ventanas.

Figura 43.

Carga de la nueva configuración



Nota. Carga de la nueva configuración. Elaboración propia, realizado con WireShark.

Con lo anterior ya se tendrá configurado WireShark para visualizar los trazados.

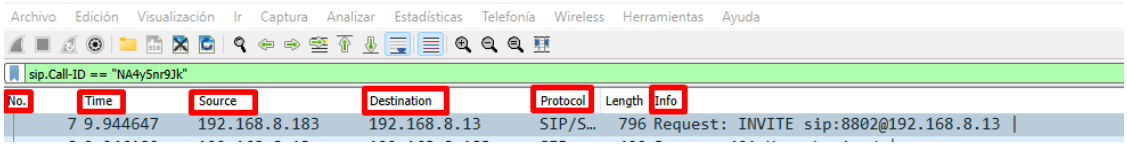
4.3.3. Pasos para analizar un trazado utilizando WireShark

Para analizar una traza se debe tener en cuenta varios factores: el tiempo, el origen, el destino, todos son importantes al momento de analizar alguna falla. Como se puede observar se tiene una IP de origen y una IP de destino, el

protocolo utilizado y la información del mensaje, en este último apartado se tendrán los tipos de mensajes para analizar.

Figura 44.

Analizar un trazado utilizando WireShark



Nota. Análisis de un trazado. Elaboración propia, realizado con WireShark.

Hay algunos casos donde, al capturar la mensajería, pueden venir contenidos, muchas llamadas o tráfico de otros UE, en WireShark se puede configurar algunos filtros para obtener la llamada que se busca obtener. Se puede darl un ctrl + B y aparecerá una nueva línea donde se podrán realizar las búsquedas, ya sea por msisdn, imsi, dirección IP, entre otros. Se debe seleccionar las opciones que más convengan, por ejemplo, si es algún msisdn se debe colocar “cadena”, “detalles de paquete” y el msisdn que se desea buscar. Con lo anterior buscará en todo el archivo PCAP dentro de cada mensaje el msisdn (File-Extension, s.f.).

Figura 45.

Obtención de llamada

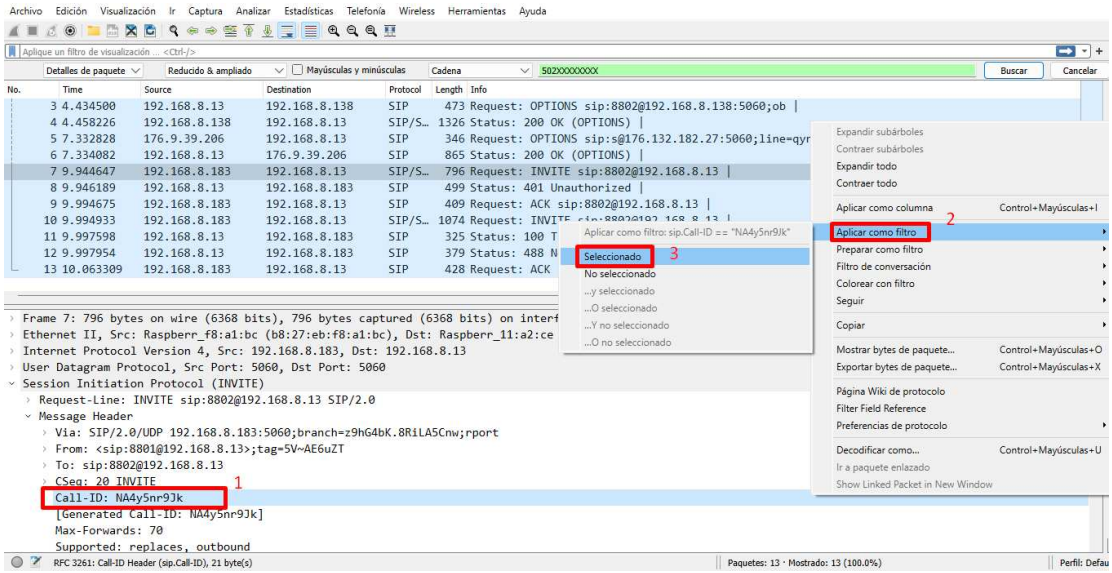
Nota. Obtención de llamada. Elaboración propia, realizado con WireShark.

Cuando se necesita reportar o escalar algún caso, hacia otro operador o cliente, se debe tener cuidado con la información que se brinda, es por esto que se deben filtrar las direcciones IP para que únicamente aparezcan las del SBC y el destino que se desea.

Para esto, si ya se tiene identificado el mensaje, se puede utilizar el parámetro *Call*-ID, el cual no cambiará en una secuencia de mensajes entre dos elementos. Primero se debe identificar el mensaje *Call*-ID, se le da clic derecho con el *mouse* y desplegará una lista de opciones. Es necesario escoger entre ellas “Aplicar como filtro” y posterior en una nueva lista de mensajes escoger “Seleccionado”.

Figura 46.

Filtrado por Call-ID

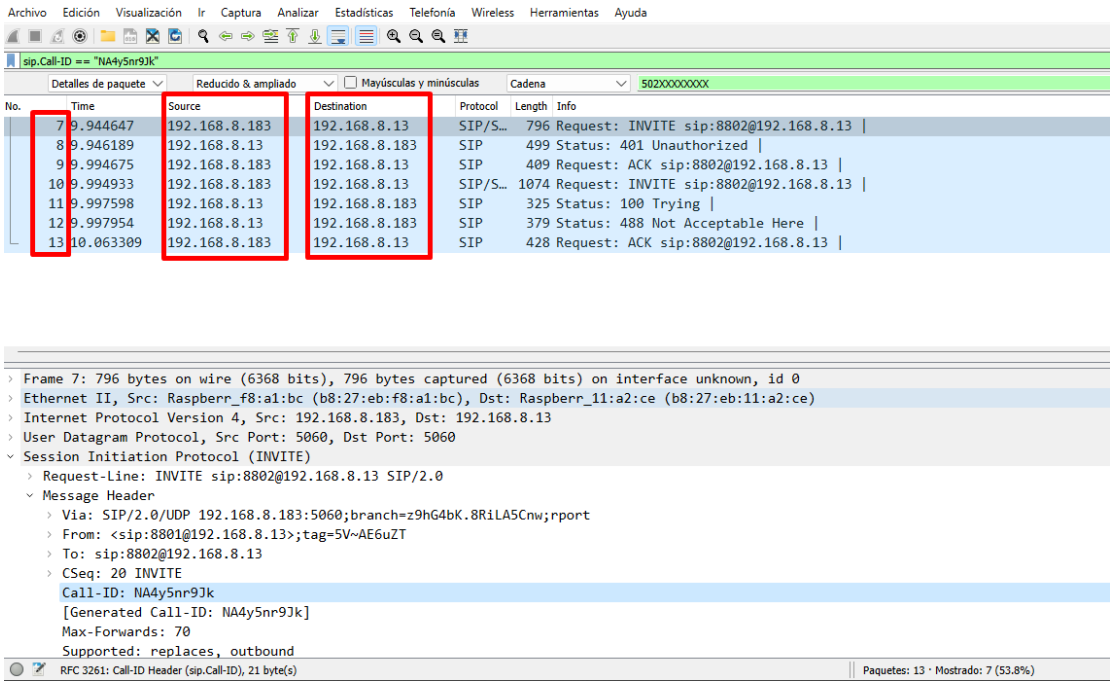


Nota. Filtrado por *Call*-ID. Elaboración propia, realizado con WireShark.

Ejecutando lo anterior se obtendrá el trazado filtrado por *Call*-ID entre las dos direcciones IP que interesan.

Figura 47.

Trazado filtrado por Call-ID

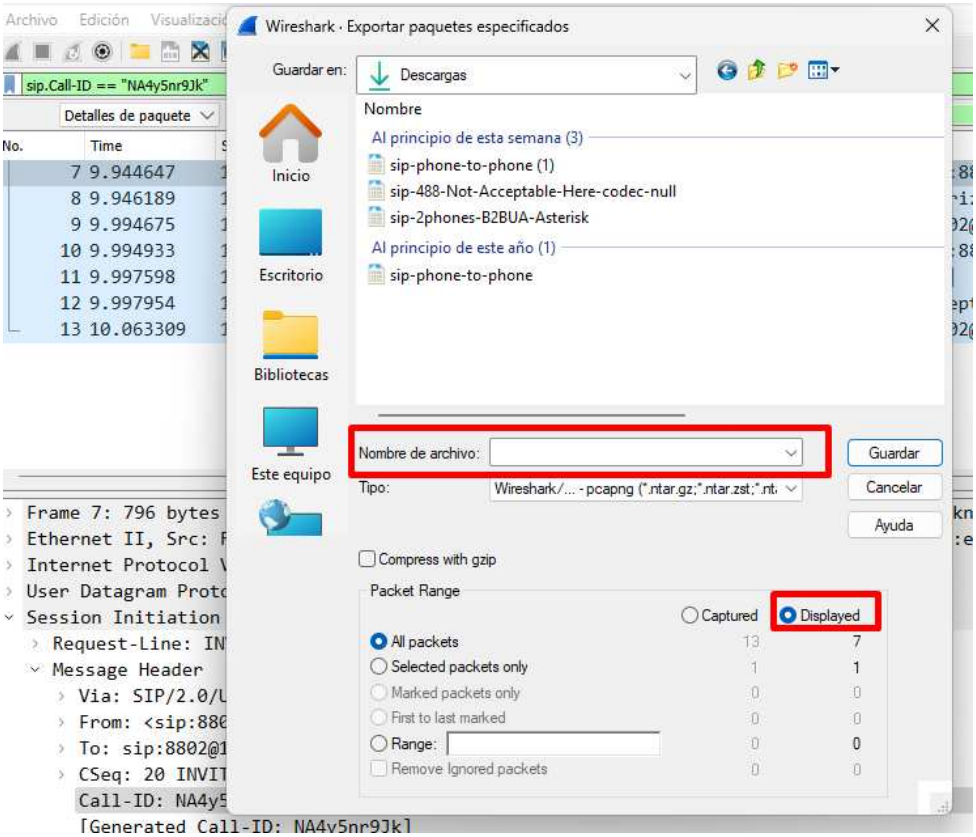


Nota. Trazado filtrado por *Call*-ID. Elaboración propia, realizado con WireShark.

Y ahora, si se puede guardar el trazado con las direcciones IP´s que interesan, en Archivo, exportar paquetes especificados y escoger la opción “todos los paquetes” y “mostrados”.

Figura 48.

Guardar trazado con dirección IP

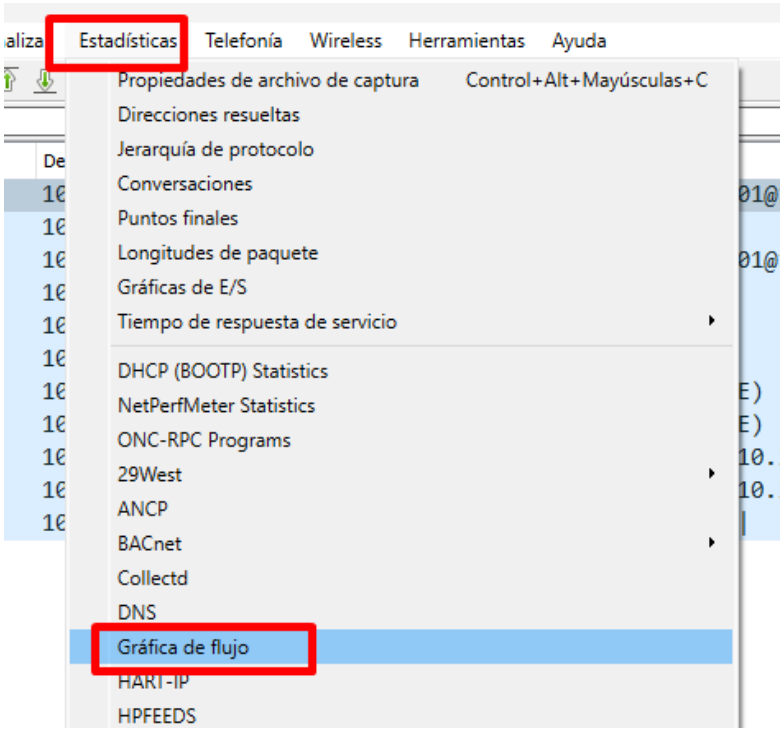


Nota. Guardar trazado con dirección IP. Elaboración propia, realizado con WireShark.

Otra de las herramientas útiles, y quizás una de las más didácticas, es ver el flujo de los mensajes de una forma gráfica. Para esto hay dos formas. La primera ir a “Estadísticas” y después a “Gráfica de flujo”.

Figura 49.

Flujo de mensajes

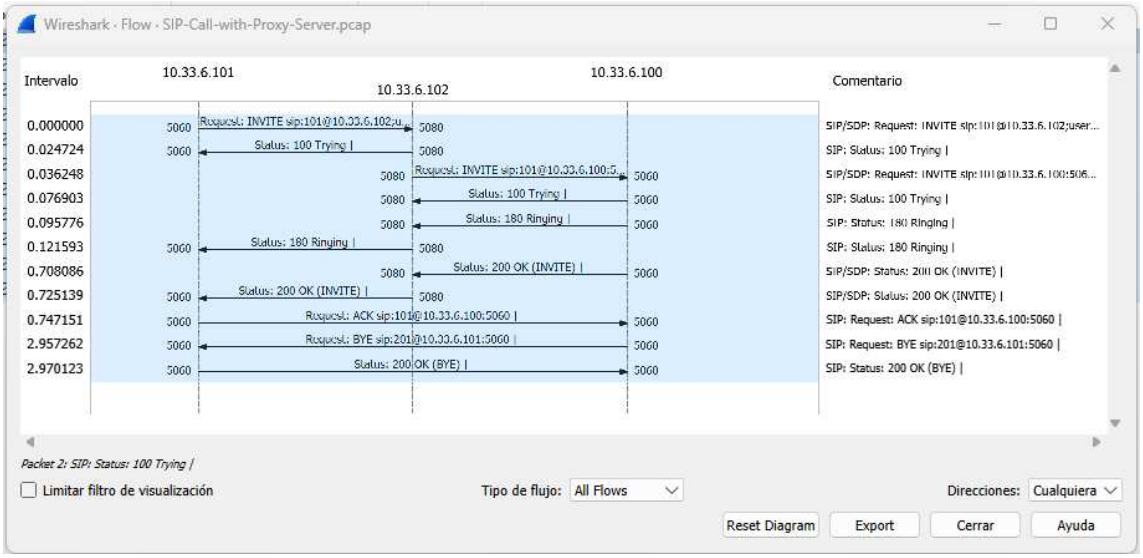


Nota. Flujo de mensajes. Elaboración propia, realizado con WireShark.

Con esto se puede observar de una forma resumida, y en cierta forma gráfica, el flujo que tuvo la llamada. Como se aprecia en la imagen, se encuentran los mensajes y las direcciones IP de los nodos.

Figura 50.

Resumen de llamadas y mensajes con direcciones IP

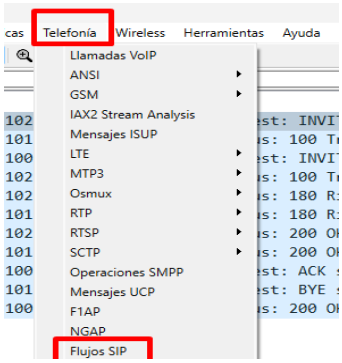


Nota. Resumen de llamadas y mensajes con IP. Elaboración propia, realizado con WireShark.

Otra forma de realizar lo anterior es, si se tiene que la llamada contiene protocolo SIP, ir a “Telefonía” en la barra de herramientas, y después seleccionar “Flujos SIP”.

Figura 51.

Revisar si la llamada contiene protocolo SIP

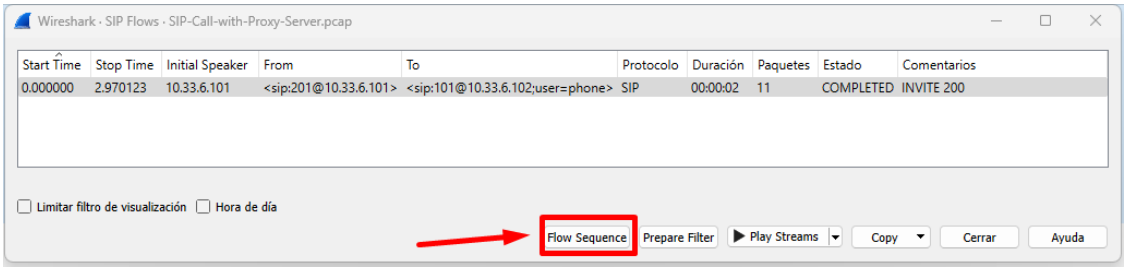


Nota. Revisar si la llamada contiene protocolo SIP. Elaboración propia, realizado con WireShark.

Realizando esto se pueden ver todos los flujos SIP que existen en el trazado, estos van a estar ordenados por From y To, es decir que, si se tienen varios mensajes de distintos orígenes y destinos, se observarán varios flujos, y será de escoger el que se está buscando, seleccionar *Flow Sequence.*

Figura 52.

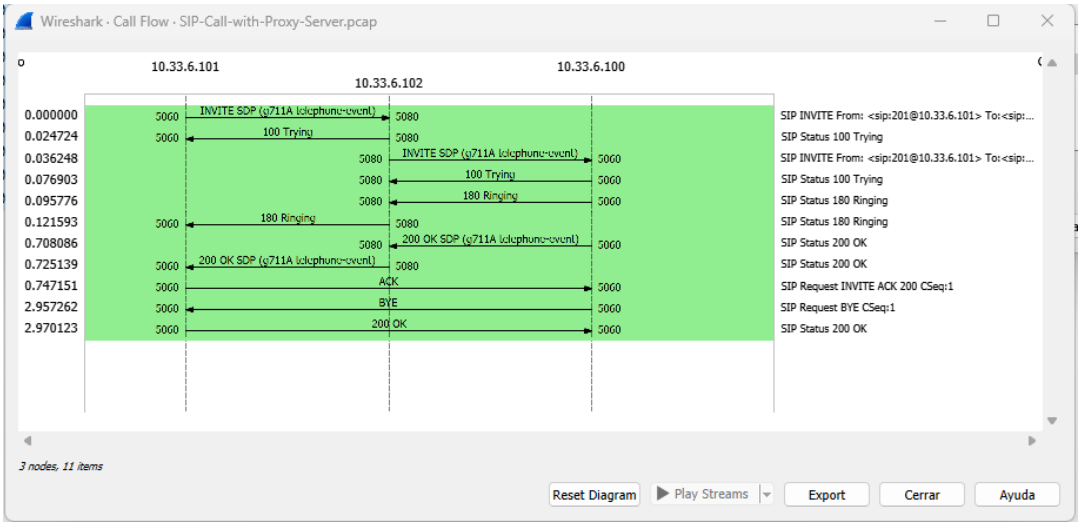
Mensajes con distintos orígenes y destinos



Nota. Mensajes con distintos orígenes y destinos. Elaboración propia, realizado con WireShark.

Figura 53.

Flujo SIP de llamada



Nota. Flujo SIP de llamada. Elaboración propia, realizado con WireShark.

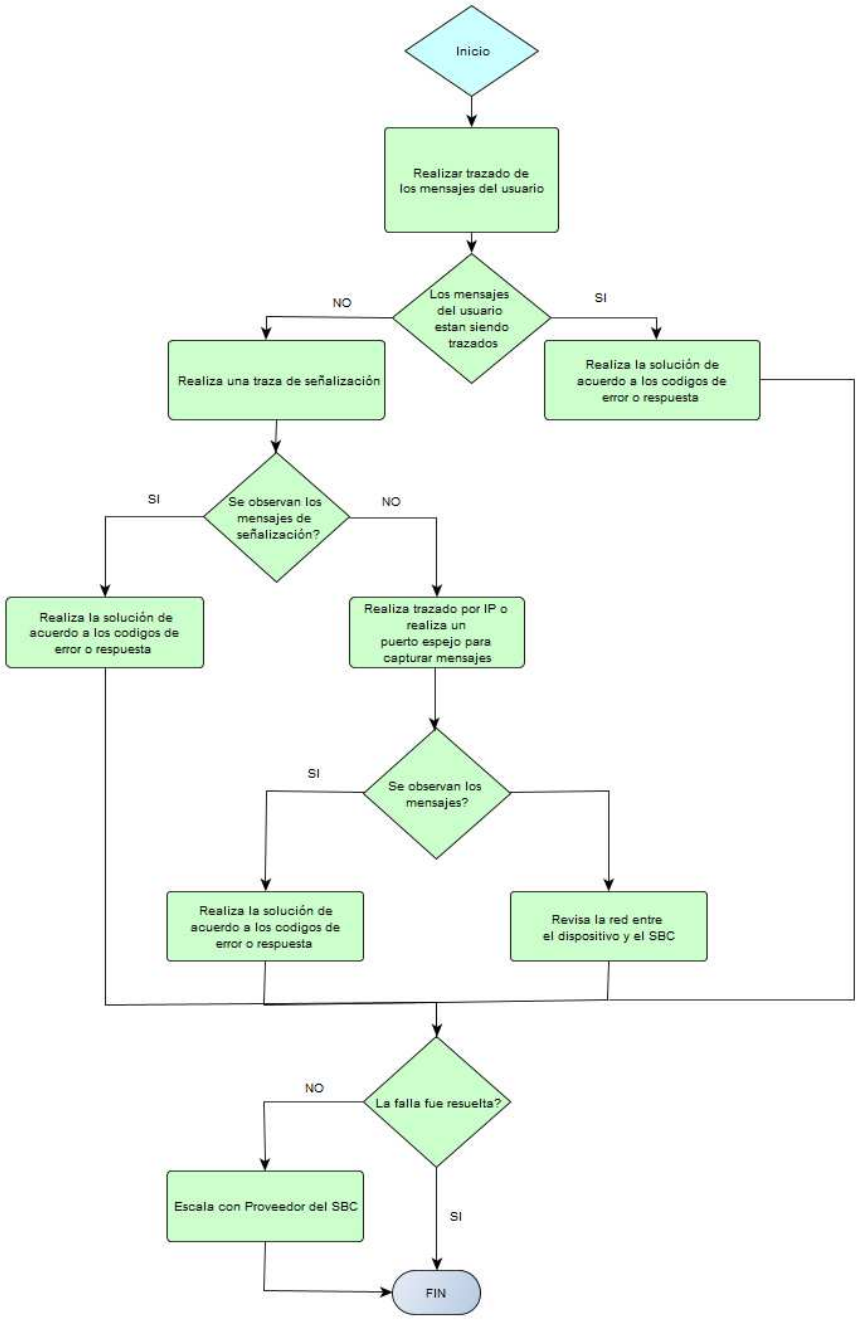
4.4. Localización de fallas de llamadas

En el análisis de problemas en llamadas es importante conocer el flujo de llamadas exitosas, cada escenario es diferente, pero si se tienen los lineamientos se hará más fácil dar con el problema.

No importando el proveedor, los SBC deben contar con métodos de capturar el tráfico, en los más completos se pueden colocar trazados por número de A o de B, por IP de la troncal, por tipo de señalización, entre otros. A continuación, se presenta un flujo que puede ayudar a visualizar los pasos a seguir al momento de realizar trazados.

Figura 54.

Localización de fallas de llamadas



Nota. Localización de fallas de llamadas. Elaboración propia, realizado con WireShark.

4.4.1. Códigos de respuesta

Anteriormente se observó los códigos de respuesta y sus características. En este apartado se verá un poco más sobre los distintos tipos de mensajes que se pueden encontrar, esto para analizar las fallas que se presentan.

4.4.1.1. Provisional 1xx

Las respuestas provisionales, también conocidas como repuestas informales, indican que el *server* al cual se le hace la solicitud está realizando una acción adicional y no tiene una respuesta definitiva para responder la solicitud. Un servidor envía un mensaje 1xx si espera tardarse más de 200ms para obtener una respuesta final. Estos 1xx pueden contener cuerpos de mensaje (sdp):

- 100 *trying:* este mensaje indica que la solicitud ha sido recibida por el servidor siguiente a SBC y que alguna acción se está tomando en referencia a la llamada. Algo importante de este mensaje es que detiene las retransmisiones de los *Invite* por parte del UAC.

- 180 *ringing:* mensaje utilizado para indicar que al usuario al que le ha llegado la solicitud (*Invite*) se le está alertando de la llamada.

- 181 *call is being forwarded:* este mensaje es utilizado para indicar que la llamada está siendo redireccionada a otro destino.

- 182 *queued:* el destino al que se ha intentado llamar está temporalmente fuera de servicio, sin embargo, el servidor ha decidido encolar la solicitud en vez de rechazarla. Cuando el destino se encuentre de nuevo disponible

transmitirá el mensaje para ser completado. En el texto del motivo puede venir más información, por ejemplo: hay 3 llamadas en cola, el tiempo esperado de espera es de 15 minutos.

- 183 *session progress:* este mensaje es utilizado para transmitir información sobre el progreso de la llamada.

4.4.1.2. Exitoso 2xx

Estos mensajes indican que la solicitud se procesó exitosamente.

- 200 *ok:* la solicitud fue exitosa.

- 202 *accepted:* mensaje que indica que la solicitud se aceptó, pero aún no puede ser procesada.

- 204 *no notification:* indica que la solicitud se aceptó pero no se recibirá mensaje de respuesta.

4.4.1.3. Redireccionamiento 3xx

Mensajes que brindan información sobre una nueva ubicación del usuario o de alguna alternativa de servicio que cumpla con la solicitud realizada.

- 300 *multiple choices:* la dirección en la solicitud es resuelta para múltiples opciones, el usuario puede seleccionar entre las opciones y redirigir la solicitud hacia esa dirección.

- 301 *moved permanently:* el usuario ya no puede ser encontrado en la dirección de la solicitud (*Request*-URI) y la solicitud debe retransmitirse a la nueva dirección que se encuentra en el campo *contact* dentro del encabezado. Para futuras solicitudes debe actualizase esta dirección y enviarse a esta nueva dirección.

- 302 *moved temporarily:* el cliente redirecciona la solicitud hacia la nueva dirección que se encuentra en el campo *contact* del encabezado.

- 380 *alternative service:* la llamada no es exitosa, pero existen otras alternativas para que complete.

4.4.1.4. Errores de solicitud 4xx

Los mensajes 4xx son mensajes de error definitivo en la solicitud que se realiza a un servidor.

- 400 *bad request:* la solicitud no ha podido ser entendida por el servidor debido a una mala sintaxis. En el texto del motivo debe indicarse más información sobre el problema.

- 401 *unauthorized:* este mensaje indica que la solicitud requiere una autenticación del usuario. Mientras que el 401 se refiere al usuario, un mensaje 407 es utilizado para los servidores *proxy.*

- 403 *forbidden:* el servidor comprende la solicitud, pero la rechaza, esto independientemente de que sea un usuario autorizado. La solicitud no se debe repetir.

- 404 *not found:* el servidor concluye que el usuario no existe en el dominio especificado por la *Request*-URI.

- 405 *method not allowed:* el método especificado en la línea de solicitud es comprendido por el servidor, pero no permitido para la dirección indicada en la *Request*-URI.

- 406: *not acceptable:* el recurso solo es capaz de generar respuestas con contenido no aceptable.

- 408 *request timeout:* el servidor no logra responder a la solicitud en un tiempo determinado. Esto se da por ejemplo cuando el servidor no logra ubicar al usuario. La solicitud se puede reenviar al servidor en cualquier tiempo.

- 415 *unsupported media type:* el servidor rechaza la solicitud del cliente porque el mensaje está en un formato que no es soportado por el servidor. El servidor debe enviar una lista de los formatos aceptados.

- 480 *temporarily unavailable:* el sistema de la persona a la que se llama fue contactado, sin embargo, actualmente no está disponible, una de las causas puede ser que se ha activado la característica “no molestar”. Este mensaje también puede darse debido a algún problema interno de la red destino.

- 481 *call / transaction does not exist:* este mensaje indica que se ha recibido una solicitud que no cumple con ningún diálogo o transacción existente.

- 482 *too many hops:* este mensaje se presenta cuando la solicitud contiene en el campo del encabezado (*header) max-forwards* con valor cero.

- 484 *address incomplete:* el servidor recibe una solicitud con una *Request*-URI que está incompleta.

- 486 *busy here:* el destinatario recibió exitosamente la solicitud, pero no está disponible o no desea atender la llamada. Este mensaje puede llevar a un buzón de voz hacia donde la llamada puede ser redirigida.

- 487 *request terminated:* la solicitud fue terminada por un mensaje *bye* o *cancel.*

- 488 *not acceptable here:* esta respuesta indica que algunos campos de la descripción de la sesión en la *Request*-URI no son aceptados.

4.4.1.5. Errores de fallo en servidor 5XX

Estos mensajes son respuestas dadas por el servidor mismo cuando ha fallado.

- 500 *server internal error:* el servidor ha encontrado una condición que evita que la solicitud sea procesada.

- 502 *bad gateway:* el servidor actuando como un *gateway* o *proxy* recibe una respuesta inválida de un servidor debajo de él (en cascada) y esto provoca que no sea posible procesar la solicitud.

- 503 *service unavailable:* este mensaje se brinda cuando el servidor está temporalmente fuera de servicio debido a un sobrecargo o mantenimiento.

- 504 *server time-out:* el servidor trató de realizar una solicitud a otro servidor, pero no hubo respuesta.

4.4.1.6. Fallas globales 6XX

Estas respuestas indican que el servidor tiene información de que el usuario no puede ser contactado.

- 600 *busy everywhere:* la solicitud fue procesada exitosamente pero el destinatario está ocupado y no desea responder la llamada.

- 603 *decline:* la solicitud fue procesada pero el usuario no desea responder o participar en la llamada.

- 604 *does not exist anywhere:* el servidor envía esta respuesta cuando tiene información de que el usuario no existe en ninguna parte.

- 606 *not acceptable:* la solicitud fue procesada pero algunos campos del SDP, es decir de la descripción de la sesión, como por ejemplo la media, el ancho de banda o el *codec*, no son aceptados. Este mensaje significa que el destino sí quiere participar en la llamada, pero no soporta adecuadamente la sesión que se quiere establecer.

4.5. Casos de fallas

A continuación, en el quinto apartado de este cuarto capítulo de la investigación, se abordarán los casos de fallas en el sistema estudiado y sus características especiales.

4.5.1. Fallas de operación

Entre los casos de fallas que es necesario examinar desde el inicio de este recorrido están las fallas que conciernen puntualmente con la operación de los servicios prestados.

4.5.1.1. Acciones por realizar si falla la troncal

Las acciones para realizar varían entre proveedores y no todos tendrán los sistemas adecuados para realizarlas, entre las más conocidas están:

- Desvío de llamadas: si la troncal primaria falla, el SBC puede ofrecer un desvío de llamadas a otro destino, ya sea móvil u otra troncal.

- Separar troncales por servicios: se pueden tener troncales dedicados, por ejemplo, para servicios críticos y para uso general, así si uno falla se tendrá el otro funcionando, y se podría redirigir el tráfico.

- Balance de carga: los SBC pueden manejar troncales con balance de carga, esto significa que el servicio puede apuntar a varias troncales y así prevenir que se sature el sistema.

4.5.2. Fallas de servicio

Además de las fallas de operación, también es necesario considerar todos los aspectos relacionados con las fallas que el sistema de servicio pueda tener directamente en su funcionamiento.

4.5.2.1. Falla de registro

- Síntoma: usuario (UE) no logra registrarse. Por ejemplo, un *softphone* instalado en alguna computadora falla al intentar registrarse a través del SBC, y al tomar trazado desde la interfaz el usuario no se observa ningún mensaje (CountherPath, 2021).

- Localización de la falla: realizar un PING desde el PC hacia el SBC, en este caso el A-SBC, o la parte de acceso. Si desde el UE la IP del SBC es alcanzable significa que el SBC está rechazando la solicitud de registro.

- Análisis de la causa: puede deberse a varios escenarios.

- Políticas de ACL, posiblemente la dirección IP del origen está negada.

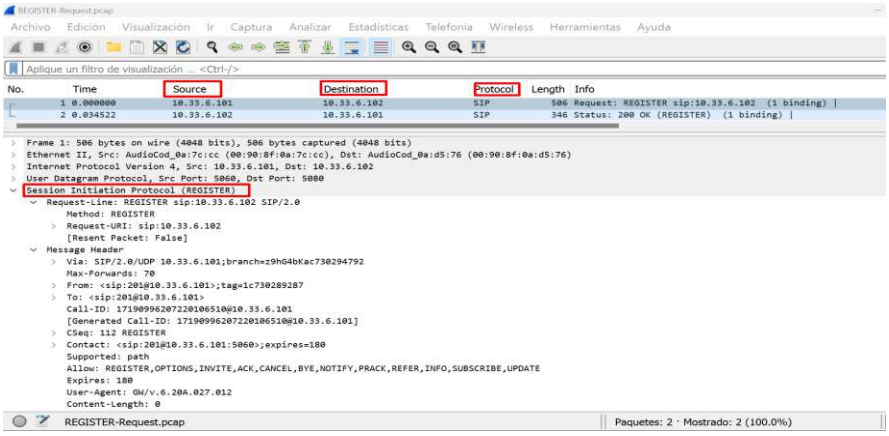
- Validar si el segmento de IP´s de origen está permitida.

- Recuperación de la falla: si alguna de las causas anteriores se da, se debe permitir la IP en el ACL o agregar el segmento IP de origen.

A continuación, se observa un trazado desde WireShark, este se tomó desde la nube, pero se debe prestar atención a cómo se presenta un trazado.

Figura 55.

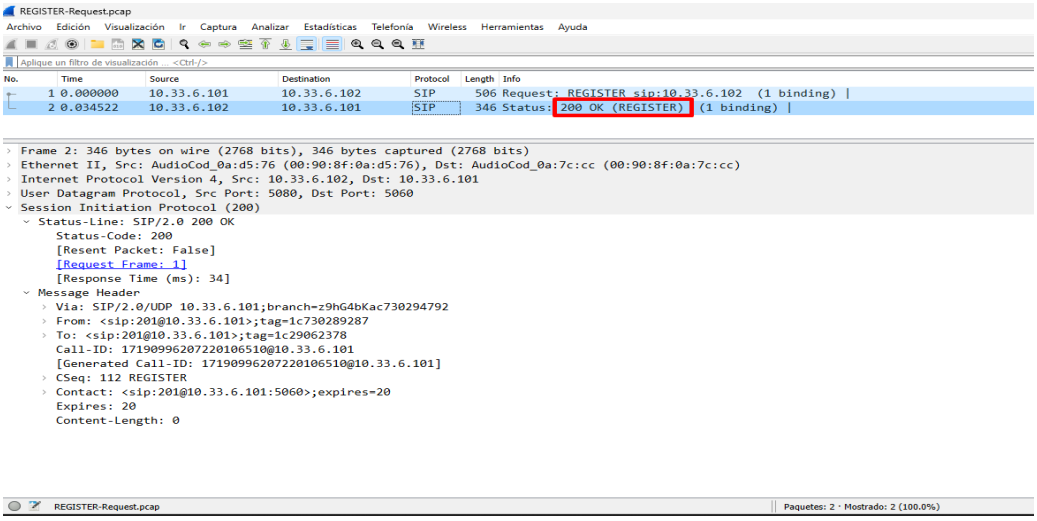
Trazado de solicitud de registro



Nota. Trazado de solicitud de registro. Elaboración propia, realizado con WireShark.

Figura 56.

Respuesta de la solicitud de registro desde el SBC



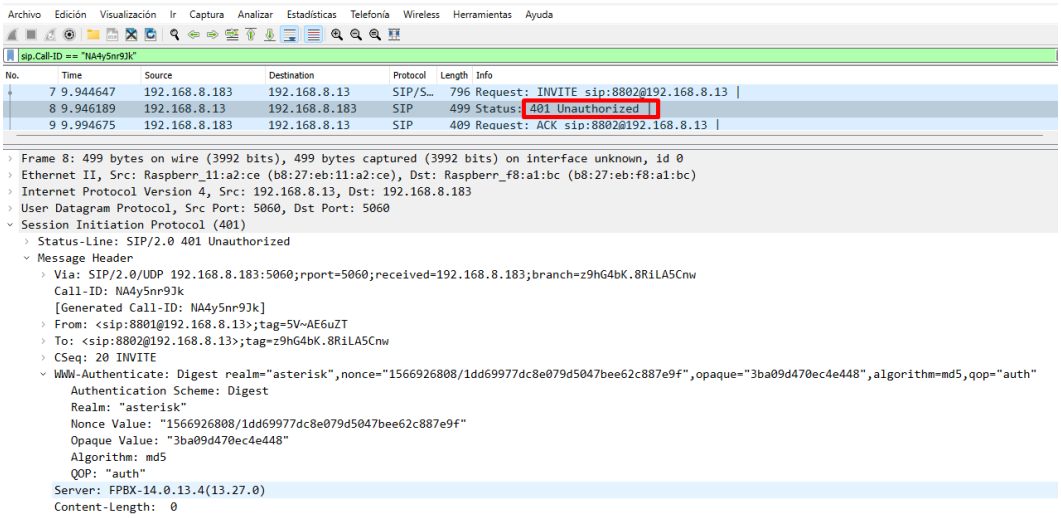
Nota. Respuesta de solicitud de registro. Elaboración propia, realizado con WireShark.

Al tomar el caso anterior se observa otro síntoma que se puede estar dando:

- Síntoma: usuario (UE) no logra registrarse. Por ejemplo, un *softphone* instalado en alguna computadora falla al intentar registrarse a través del SBC. En la traza de señalización del usuario se observa que el SBC envía la solicitud de registro a la red de *core,* el *core* responde con un mensaje SIP 401 *unauthorized,* el SBC reenvía el mensaje hacia el UE. Pero en este caso el SBC no recibe ninguna nueva solicitud de registro por parte del UE. Entonces el SBC retransmite este mensaje 401 hacia el UE.
-

Figura 57.

Usuario (UE) no logra registrarse



Nota. Usuario no logra registrarse. Elaboración propia, realizado con WireShark.

- Análisis de la causa: el UE sí envía la solicitud de registro al SBC pero los mensajes del SBC no logran llegar al UE. Al realizar una prueba de PING desde la dirección de acceso del SBC hacia el UE esta falla.

- Recuperación de la falla: se puede configurar una ruta estática para los mensajes enviados desde el SBC hacia el segmento del UE.

Hasta el momento se han visto casos de falla en registro, pero qué pasa si el registro está correcto pero la llamada no está completa, se pueden dar muchos escenarios para que una llamada no complete, desde falla en la transmisión de paquetes hasta problemas con los *codecs* de audio. A continuación, se verán unos ejemplos.

- Síntoma: usuario reporta que las llamadas no completan hacia sus equipos, puede ser una PBX.

- Análisis de la causa: al trazar en el SBC se puede ver el flujo que tiene la llamada, desde que llega el *Invite* hacia el SBC por la parte de acceso y este se envía hacia la parte del *core.* Los trazados se pueden colocar del lado de la troncal del cliente, puede ser por IP o por el número que reporta el cliente. Al colocar el trazado se observa que el SBC envía el *Invite* pero no tiene respuesta del *peer,* al realizar pruebas de PING estas son exitosas. Se debe validar con el *peer* por qué no responde a los *Invite* enviados.

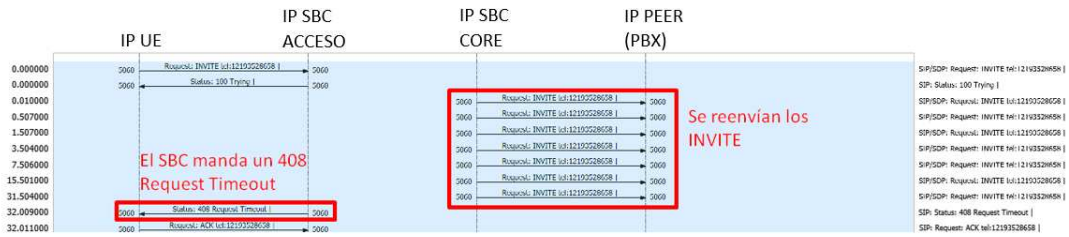
- Localización de la falla: por medio de los trazados se observa que el inconveniente está del lado del cliente. Podría tratarse también de algún problema de transmisión, sin embargo, ya que las pruebas de PING o *tracert* realizadas dan resultados exitosos, falta algo en la configuración

del cliente o pueda deberse a que la PBX se quedó inhibida y requiere de un reinicio para que levanten todos sus servicios.

- Recuperación de la falla: reinicio del lado *peer,* o con un *backup* cargar de nuevo la configuración.

Figura 58.

Casos de falla en registro



Nota. Casos de falla en registro. Elaboración propia, realizado con WireShark.

- Síntoma: se reporta que, al llamar a un destino, la llamada no completa.

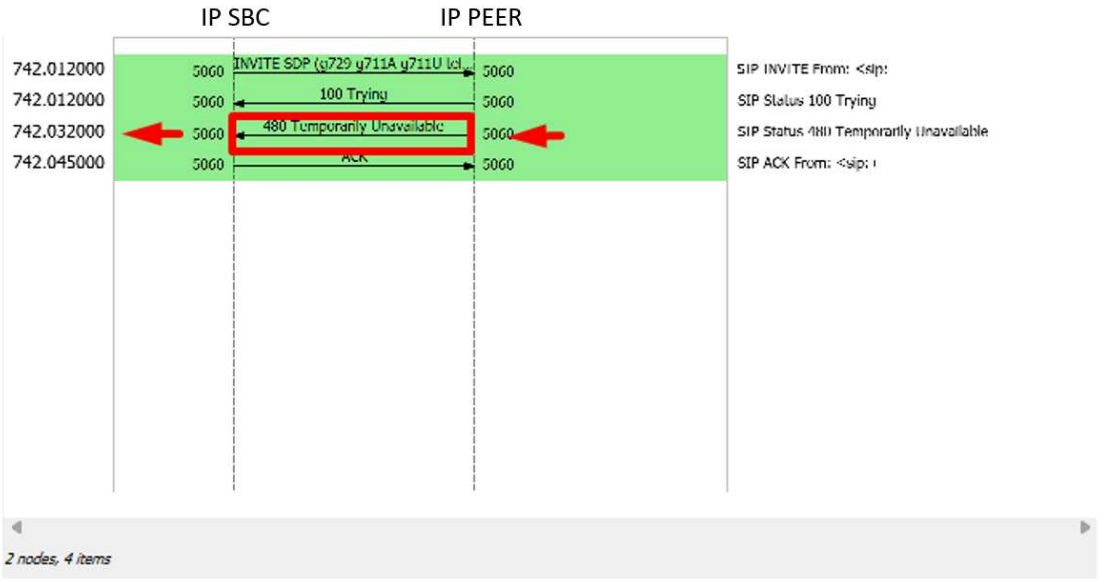
- Análisis de la causa: de nuevo se coloca un trazado en el SBC y en la mensajería se observa que el *Invite* llega desde el *core* de red, el SBC lo reenvía hacia el *peer,* sin embargo, se recibe un mensaje SIP 480 *temporarily unavailable.* Este caso se diferencia del anterior porque la llamada sí llega al destino, pero este por algún motivo no la procesa, puede deberse a un problema interno, quizás que se haya saturado el *peer,* o algún problema en los recursos.

- Localización de la falla: la falla se da del lado del *peer*, ya que el mensaje se recibe de su IP.

- Recuperación de la falla: se indica al *peer* que debe validar su equipo.

Figura 59.

Flujo del 480 es hacia el SBC



Nota. Flujo del 408 hacia el SBC. Elaboración propia, realizado con WireShark.

- Síntoma: usuarios reportan que la llamada no completa hacia un destino internacional.

- Análisis de la causa: se toma un trazado desde el SBC, como se ha mencionado el I-SBC es una interconexión entre el *core* de red y otros operadores o clientes, en este caso se trata de un *carrier,* el que, en llamadas internacionales, toma el tráfico y lo envía a su país de destino. Como puede deducirse, el *carrier* puede tener a su vez distintas rutas para entregar el tráfico hacia el país de destino, y el problema puede estar quizás en el número de destino, sin embargo, ya que el SBC le entrega el

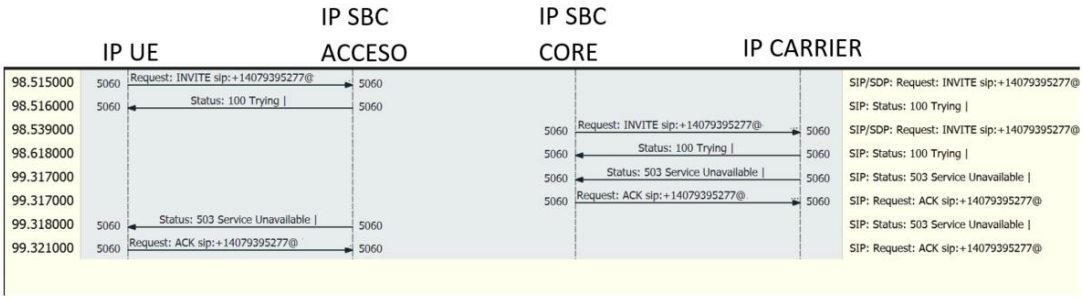
tráfico a la troncal del *carrier,* se debe hacer el reclamo a este. En este caso la llamada no completa y al validar el trazado se observa un mensaje SIP 503 *service unavailable.* Este mensaje se puede deber a que el servidor destino o la ruta no esté disponible.

- Localización de la falla: en este caso es difícil saber, puede que esté del lado del *carrier,* del país destino, o del propio número al que se quiere llamar.

- Recuperación de la falla: se reporta al *carrier,* que a su vez reporta al destino y se debe completar la llamada si es que el destino se encuentra disponible y aún dado de alta.

Figura 60.

La llamada no completa hacia un destino internacional



Nota. La llamada no completa hacia un destino internacional. Elaboración propia, realizado con WireShark.

- Síntoma: usuarios reportan no completar las llamadas hacia un destino.

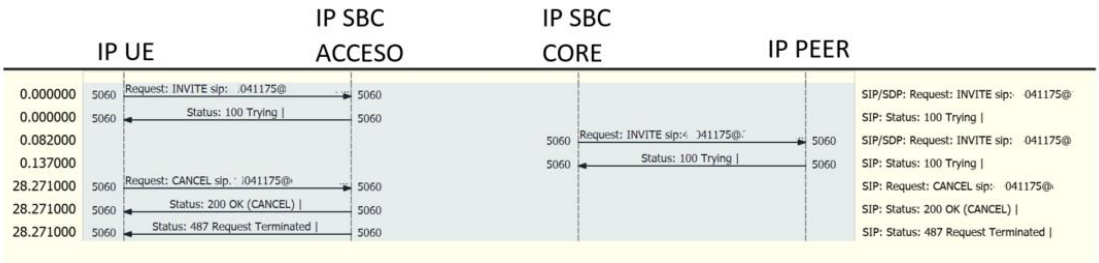
- Análisis de la causa: este caso es uno de los más comunes cuando hay problemas de llamadas, y no es por el tipo de falla a la que se debe, sino porque el usuario que reporta la falla realiza la prueba, sin embargo, él, al no escuchar nada en su auricular, cuelga o termina la llamada y, debido a esto, se observa un mensaje *cancel* al *Invite* original en la parte del acceso del SBC, el cual lo retransmite hacia el destino, y después responde al UE con un 487 *Request Terminated,* y no se puede detectar la causa de la falla, ya que no se tiene un mensaje concluyente de lo que ocurre en el *peer,* el SBC o el UE.

- Localización de la falla: no se puede determinar. Se debe solicitar nuevas pruebas, esta vez dejando que la llamada se finalice sin intervención o cortarla.

- Recuperación de la falla: al tener una traza con todo el flujo se podría analizar y ver cuál es la causa para resolverla.

Figura 61.

No completar las llamadas hacia un destino



Nota. No completar las llamadas hacia un destino. Elaboración propia, realizado con WireShark.

- Síntoma: usuarios reportan que la llamada no completa hacia un destino en específico.

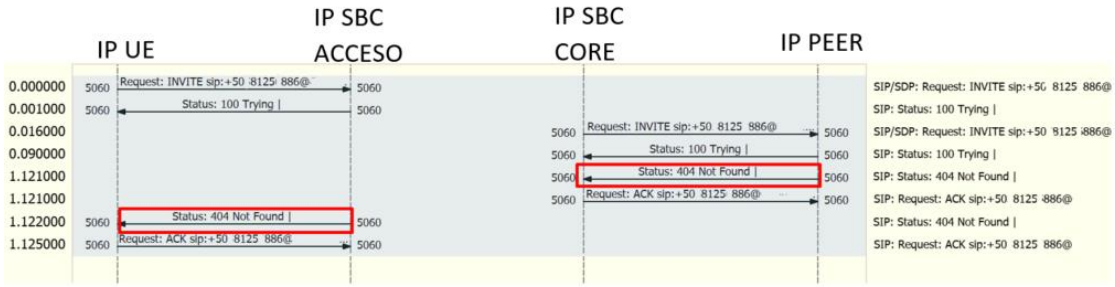
- Análisis de la causa: se coloca trazado en nuestro SBC, al validar los trazados se observa que la llamada se entrega hacia el *peer* del destino, sin embargo, se recibe un mensaje SIP 404 *not found.* Con este mensaje la llamada sí llega al destino, pero este no encuentra o no tiene el destinatario y por eso devuelve el mensaje que no lo encuentra.

- Localización de la falla: puede ser que, en efecto, el destino no se encuentre en el equipo y no sea de por sí una falla sino una respuesta correcta del *peer,* sin embargo, se debe revisar que en el *Invite* desde el UE y el SBC estén los campos correctos, y las direcciones IP coincidan con el segmento asignado para el *peer.*

- Recuperación de la falla: para este caso se debe validar los mensajes *Invite* y la causa que devuelve el 404.

Figura 62.

La llamada no completa hacia un destino en específico

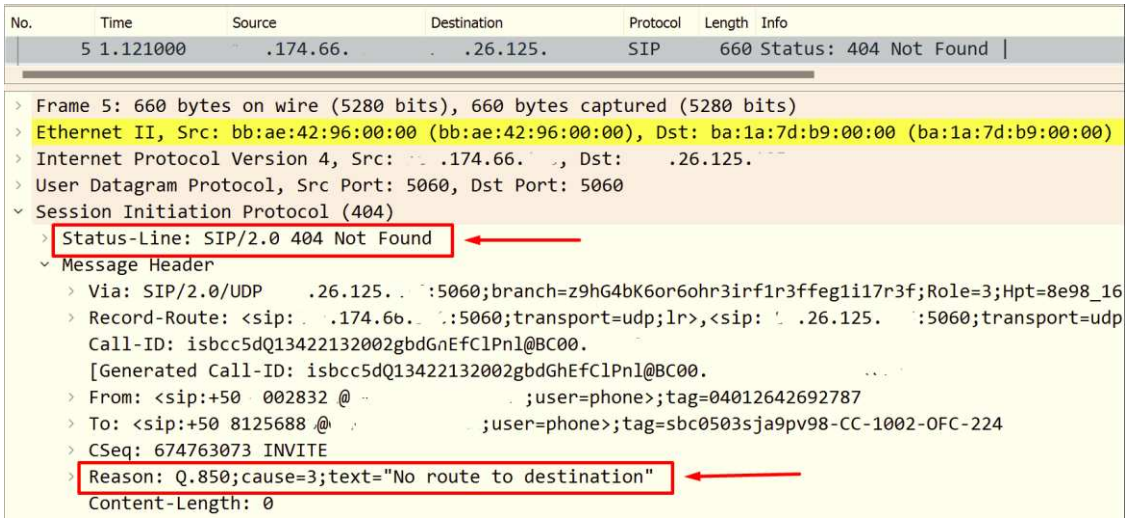


Nota. La llamada no completa hacia un destino en específico. Elaboración propia, realizado con WireShark.

Dentro del mensaje 404 se puede observar la causa *No route to destination,* puede ser indicativo de que no hay una ruta disponible para completar la llamada.

Figura 63.

No route to destination, no hay una ruta disponible para completar la llamada



Nota. No hay ruta disponible para completar la llamada. Elaboración propia, realizado con WireShark.

- Síntoma: cliente reporta que en horas de alto tráfico su troncal se queda sin realizar/recibir llamadas. Cliente realiza pruebas y observa que el SBC le libera la llamada.

- Análisis de la causa: tal como se vio en el capítulo anterior, en un ataque DoS un atacante envía un gran número de paquetes de datos a un *host* para agotar sus recursos. Al realizar trazados desde el SBC se observa que el cliente envía solicitudes, el SBC las procesa y las envía hacia el

core, pero justo antes de que el *core* responda con el 200 *ok* el cliente cuelga la llamada, esto resulta en una llamada incompleta.

- Localización de la falla: sin embargo, suponiendo que el cliente es un *call center,* que manejará mucho tráfico, se puede dar la condición en la cual los operadores realizan llamadas, pero no esperan lo suficiente para que la llamada complete, por el hecho de querer contactar a más clientes, ocasionando que en un corto tiempo se den muchas llamadas incompletas, el SBC detecta este tipo de comportamiento y se protege haciendo uso de sus políticas, bloqueando la IP que origina este tráfico y colocándola en una lista negra.

- Recuperación de la falla: se deberá validar con el cliente la forma de reducir al máximo este comportamiento, y una solución definitiva, ya que se conoce el origen de estos eventos, es colocar la IP del cliente en una lista blanca, para que así, aunque se den los eventos, no se interfiera el servicio del cliente.

CONCLUSIONES

1. Con este trabajo se ha descrito el funcionamiento general de un controlador de sesión de frontera utilizando el protocolo SIP y su aporte a la seguridad en una red de telefonía.

2. Se complementa a los profesionales relacionados a la rama de telecomunicaciones con conocimientos sobre el funcionamiento de la telefonía sobre IP.

3. A través de la descripción de la teoría relacionada con el controlador de sesión de frontera, utilizando el protocolo SIP, este documento es un complemento ideal para los profesionales que quieran conocer más acerca de la red de telefonía sobre IP.

4. Se brindan las bases necesarias para que los profesionales de la rama de telecomunicaciones puedan complementar sus conocimientos en las redes de telefonía sobre IP.

5. Ha sido posible describir el protocolo SIP y su aplicación en un controlador de sesión de frontera que maneja este protocolo.

6. Se han brindado casos prácticos de fallas, el procedimiento sugerido y las posibles soluciones, en una red de telefonía sobre IP.

7. Un controlador de sesión de frontera es más que un *firewall,* es un elemento indispensable hoy en día para brindar seguridad a las comunicaciones en una red de telefonía que maneja protocolo SIP.

RECOMENDACIONES

1. Recordar, al profesional que recién se está integrando al campo de las telecomunicaciones, que debe tomar en cuenta el funcionamiento de una red de telefonía para identificar los principales componentes de esta.

2. Verificar las actualizaciones que las organizaciones realizan sobre los protocolos y funciones de los equipos que se implementan en la red.

3. Verificar que se cumplan a tiempo las tareas de operación y mantenimiento en un equipo importante como el controlador de sesión de frontera.

REFERENCIAS

CISCO. (s.f.). *Teléfonos IP cisco*. CISCO. https://www.cisco.com/c/en/us/products/collaboration-endpoints/ip-phones/index.html

CounterPath. (19 de septiembre de 2021). *Softphone X-Lite isn't gone, it just has a new name – and a lot more features*. CounterPath. https://www.counterpath.com/x-lite/

File-Extension. (s.f.). *Extensión de archivo PCAP*. File-Extension. https://www.file-extension.info/es/format/pcap

Handley, M., Perkins, C. y Whelan, E. (2 de marzo de 2013). *Session announcement protocol, SAP RFC 2974*. Datatracker. https://datatracker.ietf.org/doc/rfc2974/

Huawei Technologies. (2017). *Manual de operación y mantenimiento*. Telefónica Móviles Guatemala, S.A.

Instituto Nacional de Ciberseguridad. (s.f.). *Wireshark, herramientas gratuitas*. INCIBE. https://www.incibe.es/incibe-cert/blog/analizadores-red-sistemas-control

Netacad. (29 de agosto de 2005). *Curso CCNA Exploration 4.0. Routing protocols and concepts*. Netacad. https://www.netacad.com/es

Rodger, E. y Tranter, W. (2014). *Principles of comunications. Systems, Modulation and Noise.* Wiley.

Schulzrinne, H. (30 de julio de 2003). *RTP: A Transport Protocol for Real-Time Applications*. Datatracker. https://datatracker.ietf.org/doc/html/rfc3550

Schulzrinne, H. (19 de julio de 2006). *SDP: Session Description Protocol*. Datatracker. https://datatracker.ietf.org/doc/html/rfc4566

VOIP-INFO. (8 de septiembre de 2005). *Codecs, VoIP. Get 3CX Free.* VOIP-INFO. https://www.voip-info.org/codecs/

WireShark. (s.f.). *The world's most popular network protocol analyzer.* WireShark. https://www.wireshark.org/