ANÁLISIS DEL RENDIMIENTO DE VOZ SOBRE IPV4 E IPV6, REALIZANDO MEDICIONES DEL JITTER Y DELAY EN SISTEMAS MPLS Y BGP.

Autor:
JUAN SEBASTIÁN RAMÍREZ MOSCOSO
Estudiante de Ingeniería de Telecomunicaciones

PROYECTO DE GRADO

UNIVERSIDAD SANTO TOMÁS
FACULTAD DE INGENIERÍA DE TELECOMUNICACIONES
BOGOTÁ D.C.
2014

ANÁLISIS DEL RENDIMIENTO DE VOZ SOBRE IPV4 E IPV6, REALIZANDO MEDICIONES DEL JITTER Y DELAY EN SISTEMAS MPLS Y BGP.

Autor:

JUAN SEBASTIÁN RAMÍREZ MOSCOSO
Estudiante de Ingeniería de Telecomunicaciones

Directora:

MAYRA LILIANA SALCEDO GONZÁLEZ
Ingeniera de telecomunicaciones
Master Oficial en Tecnologías, Sistemas y Redes de Comunicaciones

Asesor:

JULIO ERNESTO SUÁREZ PÁEZ
Ingeniero de telecomunicaciones
Magister en Ingeniería área Telecomunicaciones

UNIVERSIDAD SANTO TOMÁS
FACULTAD DE INGENIERÍA DE TELECOMUNICACIONES
BOGOTÁ D.C.
2014

Nota de Aceptación

_____

_____

_____

_____

_____

_____

_____
Firma del presidente del jurado

_____
Firma del jurado

_____
Firma del jurado

BOGOTÁ D.C. 2014

DEDICATORIA

Este trabajo de grado se lo dedico a Dios por darme la oportunidad de estudiar una carrera profesional, también se lo dedico a mi madre, quien es la persona más importante en mi vida, quien con sus esfuerzos me formó como una persona honesta y luchadora, ella ha sido el motor de mi inspiración para terminar mi carrera profesional, por la que quiero salir adelante brindándole un mejor futuro, en tercer lugar se lo dedico a mi padre quien me apoyo económicamente en toda mi carrera profesional preocupándose porque alcanzara mis sueños para convertirme en un profesional íntegro, muchas gracias.

AGRADECIMIENTOS

Agradecer principalmente a Dios quien me dio la fuerza para llegar hasta este punto de mi carrera profesional, cruzando obstáculos sin dejarme desfallecer. También agradezco a todos los profesores que me encontré en el camino, quienes aparte de transmitirme conocimientos me dejaron enseñanzas de vida, por ultimo agradezco a dos personas que han sido muy importantes para la culminación de mi carrera, como es la ingeniera Mayra Liliana Salcedo González y el ingeniero Julio Ernesto Suarez Páez, de los cuales estoy muy agradecido por dejarme pertenecer a este proyecto donde he obtenido nuevos conocimientos y he puesto en práctica los aprendidos durante mi carrera.

TABLA DE CONTENIDO

LISTA DE TABLAS

LISTA DE FIGURAS

GLOSARIO

ANCHO DE BANDA: Capacidad de disponibilidad de recursos en Hertz (Hz) de un canal de comunicación.

CABECERA: Información de control al principio de un mensaje, segmento, datagrama, paquete o bloque de datos.

CODEC: Codificador-Decodificador.

CODIFICACIÓN: Método para convertir un flujo de datos en otro sistema de flujo de datos.

DIRECCIÓN IP: Dirección de origen o de destino de cuatro octetos (32 bits) para identificar un pc, router u otro componente de la red.

ENRUTAR: Técnica de direccionamiento de paquetes en las redes de datos.

ESPECTRO ELECTROMAGNÉTICO: Distribución de ondas electromagnéticas en el espacio. Radiación electromagnética.

ESPECTRO RADIOELÉCTRICO: Porción del espectro electromagnético designado para las aplicaciones y tecnologías de radiocomunicaciones.

HOST: Anfitrión, referente a un ordenador conectado a las redes de datos y que hace parte integral de las mismas y de sus aplicaciones.

INTERFAZ: Periféricos o adaptadores de red que permiten la comunicación física entre dispositivos interconectados.

MULTICAST: Multidifusión, envío de paquetes de datos entre múltiples dispositivos determinados sistemáticamente.

MULTIMEDIA: Tecnologías que permiten la integración de datos de texto, imágenes y sonidos de alto nivel de manera interactiva.

NODO: Dispositivos de una red de datos que la conforman como puntos de destino u origen de flujos de datos.

PAQUETES: Bloques de datos en los que se divide la información transmitida a través de la red.

PING: Packet Internet Groper. Buscador de Paquetes de Red, programa que comprueba el estado de disponibilidad de los Host que conforman una red.

RED: Conjunto de dos o más nodos interconectados por medio de diversos sistemas, métodos y tecnologías.

RFC: Petición de comentarios (*Request For Comments*), serie de estándares de tecnologías relacionadas con internet que comenzaron a publicarse a partir de 1969.

RTP: Protocolo que se encarga de los paquetes multimedia (voz, video) transmitidos en tiempo real.

TCP: Protocolo de Control de Transmisión, responsable de la comunicación fiable de host a host en las redes de datos.

TCP/IP: Familia de protocolos de redes de datos definidos por el pentágono para definir la estructura base en la que funciona internet.

UDP: Protocolo de Datagrama de Usuario, tecnología de capa de transporte encargado de transmitir datagramas fragmentados a través de la red. Tecnología no orientada a la conexión

I. RESUMEN

A lo largo de la historia, las comunicaciones se han convertido en parte esencial de la sociedad, por lo cual la industria ha desarrollado nuevas tecnologías en el ámbito de las telecomunicaciones, dando paso a nuevos servicios como el de voz de IP (VoIP) el cual brinda diferentes características, como la disminución de costo en la implementación del servicio de telefonía, interoperabilidad con las redes de telefonía conmutadas y funcionalidades como la video llamada.

En el presente documento se hace un análisis del rendimiento de Voz sobre IPv4 e IPv6 utilizando como método de convivencia un túnel manual IPv6-IPv4, realizando mediciones del jitter y delay en sistemas MPLS y BGP. Este análisis se llevará a cabo mediante experimentos que permitan observar el comportamiento del tráfico de voz, para así posteriormente realizar las conclusiones de los experimentos realizados.

Palabras claves

Voz sobre IPv4, Voz sobre IPv6, Ubuntu, Asterisk, softphone, SIP, troncales, MPLS, BGP.

II. INTRODUCCIÓN

A través del tiempo, el crecimiento exponencial de las redes y usuarios están llevando al agotamiento de las direcciones IPv4, por lo cual se han buscado soluciones alternas como la creación de un nuevo direccionamiento: IPv6, este tipo de direccionamiento está representado con 128 bits, esto proporciona una cantidad de direcciones de 340 sextillones posibles, también brinda nuevas características como múltiples beneficios en seguridad, manejo de calidad de servicio, una mayor capacidad de transmisión y mejora la facilidad de administración, entre otros (Mejía, 2011).

Por otra parte, una de las arquitecturas más utilizadas por los operadores de telecomunicaciones es MPLS (Multi-Protocol Label Switching) el cual brinda una combinación entre velocidad e inteligencia de enrutamiento (Daisaku Shimazaki, october 2008). Este tipo de red ofrece cualidades interesantes como altos niveles de rendimiento, priorizando el tráfico que se envía por la red, por otro lado permite el flujo de voz y tráfico multimedia.

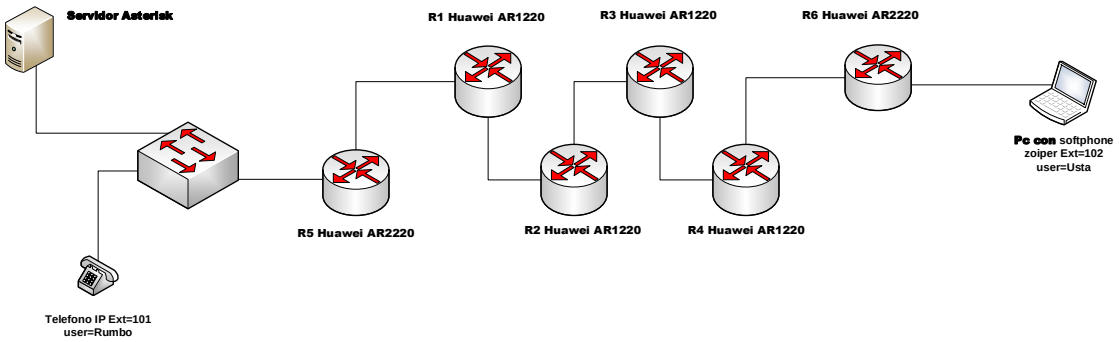
Otro aspecto relevante a tratar en este proyecto son las redes basadas en BGP pues este protocolo es la base del enrutamiento de las redes públicas (internet). BGP es el protocolo de encaminamiento entre sistemas autónomos. Un Sistema Autónomo, en inglés Autonomous System (AS), se conforma por dispositivos de red (routers) que son controlados por una sola autoridad para realizar ruteo con una única política de enrutamiento y es utilizado para comunicar los grandes nodos de Internet y transferir una gran cantidad de información entre dos puntos de la Red. Su misión es encontrar el camino más eficiente entre los nodos para propiciar una correcta circulación de la información en Internet. (Tran Cong Hung Posts & Telecommun. Technol. Inst., Thang, & Huy, 12-14 Feb. 2007)

Todas las tecnologías anteriormente mencionadas son fundamentales para el correcto funcionamiento de las redes de telecomunicaciones basadas en IP, sin embargo, el objeto de estas redes no es el uso de los protocolos, si no prestar servicios entre los cuales se destaca el servicio de voz sobre IP el cual debe funcionar independiente a la tecnología de enrutamiento o direccionamiento utilizada. Este proyecto pretende realizar un análisis del comportamiento de voz IP sobre IPv4 e IPv6 utilizando como método de convivencia un túnel manual IPv6-IPv4, realizando experimentos que permitan medir el jitter y delay en los sistemas MPLS y BGP.

Se diseñan una serie de experimentos para poder comparar el rendimiento de la voz IPv4 e IPv6 sobre sistemas MPLS y BGP. Para los primeros escenarios se realizan experimentos de forma real mediante routers Huawei ubicados en los laboratorios de la universidad, para los segundos escenarios se realizan experimentos de forma emulada mediante el software GNs3, con lo que se obtendrán datos de jitter y delay que se analizarán y posteriormente se compararán entre sí, resaltando el rendimiento de la voz sobre método de convivencia de IPv6 sobre IPv4.

A continuación se muestra la estructura que se va a emplear en los experimentos para implementar los sistemas MPLS y BGP tanto de forma real como de forma emulada para brindar el servicio de voz IP.

Ilustración 1. Estructura de experimentos.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Como se observa en la figura.1 la infraestructura de hardware con la que se cuenta, se compone de: cuatro router Huawei (AR1220) y dos router Huawei (AR2220), también se implementara la estructura en software donde se utilizará GNS3 para la emulación de las estructuras, utilizando máquina virtual (vitualbox) , también se utiliza softphone zoiper esto es necesario para realizar los experimentos diseñados, en seguida se realizarán los protocolos de pruebas necesarios para llevar a cabo los experimentos diseñados y obtener datos experimentales para su posterior análisis, los datos se obtendrán mediante el software Wireshark.

III. PLANTEAMIENTO DEL PROBLEMA

La necesidad de la sociedad de estar comunicada siempre ha sido un reto para las tecnologías. Esta se ha suplido gracias al desarrollo e implantación de redes de telecomunicaciones, teniendo en cuenta que siempre se ha contado con problemas de convergencia en estos sistemas, para que todos los servicios e infraestructura estén conectados entre sí.

En la actualidad el servicio de vos IP (VoIP por su acrónimo en inglés) ha tomado gran acogida tanto en entornos empresarias como en entornos comerciales, donde las empresas prestadoras de servicio de voz IP deben brindarlo con una buena calidad, por lo mencionado anteriormente se busca en este proyecto de investigación realizar una serie de experimentos donde se implemente una red basada en MPLS y una red basada en BGP para brindar un servicio de voz IP nativo tanto en IPv4 como en IPv6, midiendo y analizando el jitter y delay ya que son variables que directamente afectan el servicio de voz IP. Este proyecto también tiene en cuenta en segundo plano el agotamiento del direccionamiento IPv4 lo que ha hecho que las redes empiecen a migrar al direccionamiento IPv6, por tal motivo se realizarán unas recomendaciones sobre el funcionamiento del servicio de voz IPv6 mediante el método de convivencia del túnel.

De igual manera se desarrollarán una serie de experimentos donde otros autores no se hayan basado en los sistemas MPLS y BGP para brindar un servicio de voz IP en IPv4 e IPv6.

Teniendo en cuenta lo anterior la pregunta problema que se pretende resolver es: **¿cuál es el impacto con respecto a jitter y delay al implementar IPv4 e IPv6 en un servicio de voz en sistemas MPLS y BGP?.**

De acuerdo con la pregunta problema planteada, el proyecto se orienta al análisis del comportamiento de voz IP sobre IPv4 e IPv6 utilizando como método de convivencia un túnel manual IPv6-IPv4, mediante experimentos para realizar mediciones del jitter y delay en sistemas MPLS y BGP.

IV. JUSTIFICACIÓN

Este proyecto se justifica entonces en que en la actualidad las comunicaciones son indispensables en los entornos empresariales, por lo cual uno de los servicios más implementados es la voz sobre IP (VoIP por su acrónimo en inglés). Conociendo que el servicio de voz IP cada día toma más auge, se debe brindar este de forma confiable y con alta calidad, por lo cual es importante tener en cuenta varios aspectos relevantes como el tipo de red y protocolo de direccionamiento, por esto se pretende analizar la voz en sistemas MPLS y BGP configuradas tanto en IPv4 e IPv6, para implementar un servicio de VoIP (IPv4 e IPv6) y así realizar pruebas, mediciones y analizar los resultado obtenidos.

Este proyecto analizará el servicio de voz IP sobre los escenarios anteriormente planteados lo que dará herramientas para una mejor implementación y apropiación del servicio de voz sobre IPv6 en las redes de telecomunicaciones futuras.

La factibilidad operativa

El proyecto cuenta con buenos equipos para realizar los experimentos, estos equipos están ubicados en los laboratorios de la universidad y en RUMBO, esto dando paso a la implementación del servicio de voz IP tanto en IPv4 como IPv6, en los sistemas MPLS y BGP, logrando unas recomendaciones al agotamiento de las direcciones IPv4.

La factibilidad económica

El desarrollo de este proyecto cuentas con todos los equipos necesarios para poder cumplir con los objetivos planteados, los cuales son suministrados por la universidad Santo Tomás y el semillero de investigación RUMBO, también se realizarán experimentos mediante emulaciones logrando así un desarrollo del proyecto sin muchos costos.

La factibilidad tecnológica

El servicio de voz IP en IPv6 sobre los sistemas MPLS y BGP busca encontrar una solución a los problemas de agotamiento de las direcciones IP que se están presentando en la actualidad, y también dar un mejor rendimiento en los entornos donde se utiliza este servicio como entornos empresariales.

Enfoque humanista

Es importante resaltar que el proyecto de investigación análisis del rendimiento de tráfico de voz sobre IPv4 e IPv6, realizando mediciones del jitter y delay en sistemas MPLS y BGP busca prestar una solución viable a la sociedad en el entorno de comunicación vía IP, ya que esta tecnología se implementa en la actualidad en IPv4 teniendo una gran demanda, teniendo como base uno de los principios de Santo Tomás de Aquino, desarrollando conceptos que ayuden a un mejor bienestar de la sociedad.

1. OBJETIVOS

a. Objetivo general

- Analizar el rendimiento de Voz sobre IPv4 e IPv6 utilizando como método de convivencia un túnel manual IPv6-IPv4, realizando mediciones del jitter y delay en sistemas MPLS y BGP.

b. Objetivos específicos

- Diseñar experimentos que permitan analizar el rendimiento de voz sobre IPv4 e IPv6 (utilizando como método de convivencia un túnel manual IPv6-IPv4), para medir el jitter y el delay en sistemas MPLS y BGP

- Implementar un servidor Asterisk con el protocolo IPv4 e IPv6, para obtener servicios de voz sobre IPv4 e IPv6.

- Realizar un análisis del comportamiento del sistema MPLS y del sistema BGP, realizando los experimentos previamente diseñados.

- Analizar los datos de jitter y delay obtenidos de manera experimental para identificar el comportamiento de voz sobre IPv4 e IPv6 en sistemas MPLS y BGP, utilizando como método de convivencia un túnel manual IPv6-IPv4.

- Identificar el efecto sobre el jitter y delay en voz IPv6 al utilizar el método de convivencia de túnel manual IPv6-IPv4 con base a los datos obtenidos en los experimentos realizados en sistemas MPLS y BGP.

2. ESTADO DEL ARTE

2.1 ANTECEDENTES

De acuerdo con la investigación realizada, se consideró que los siguientes artículos son los más actuales y afines con el desarrollo de temática que aquí se trata, sin embargo ninguno con los mismos experimentos planteados que se pretenden realizar en este proyecto.

From voice over IPv4 to voice over IPv6 networking migration realizado por Ingeniero de Sistemas Octavio Salcedo, ingeniero Electrónico Danilo López, Ingeniero de Sistemas Fausto Alexánder Gamboa Quiroga (Bogotá 28 noviembre del 2011) Donde desarrollan un artículo de investigación de la migración del servicio de voz IP de IPv4 a IPv6. Realizan pruebas sobre una red comformada por cuatro router, un cliente #1 con sistema operativo Windows XP, un cliente #2 con sistema operativo Linux, un generador de trafico MGEN y un equipo con Wireshark para realizar la captura de tráfico. Las pruebas realizadas en el artículo consisten en llamadas de un cliente al otro sin tráfico de fondo y con tráfico, para así observar el rendimiento de la voz en los distintos escenarios implementados. Obtuvieron como resultado que en la versión 6 del protocolo IP la voz tiene un mejor funcionamiento reduciendo entre un 35% a un 50% el retardo y el jitter, también observaron en las capturas que la perdida de paquetes era diferente en los dos sistemas operativos utilizados, teniendo un valor más elevado de perdidas el sistema operativo Linux con un 24% sobre un 17% del sistema operativo Windows. Las conclusiones del artículo infieren que es importante migrar el servicio de voz IPV4 a IPv6 ya que se cuenta con más número de direcciones para signar, beneficios de costos bajos y de múltiples servicio multimedia con los que no cuenta o son costosos la red telefónica publica conmuta, mencionan que el protocolo SIP es el óptimo para manejar la señalización debía a que maneja los dos protocolos de IP. (Octavio Salcedo, 2012)

Integrating Voice over IP Services in IPv4 and IPv6 Networks realizado por Lambrinos, L. Dept. of Commun. & Internet Studies, Cyprus Univ. of Technol., Limassol, Kirstein, P., (Lambrinos, 4-9 March 2007) Done desarrollan un artículo de investigación de la interoperabilidad de las redes IPv4 e IPv6 para brindar un servicio de voz IP tanto en IPv4 como en IPv6. Realizan pruebas de en tres redes distintas conectadas entres si, donde estaban distribuidas de la siguiente manera: un servidor de voz IP para una red IPv4 donde los números de marcación empiezan por el # 4, un servidor de voz IP para un red IPv6 y un servidor para un red IPv4/IPv6 donde los números de marcación empiezan por el # 6, esto con el fin de realizar llamadas entre las redes y observar la interoperabilidad. En el artículo concluyen que el servicio de voz IP va hacer unos de los mayores impulsadores de la migración

a IPv6, se menciona que la red de internet va a durar algún tiempo con IPv6 por tal razón se necesita que las las aplicaciones migren a IPv6 y tenga una interoperabilidad. Se hace una recomendación a los fabricantes de teléfonos IP y softphone IP para integren la funcionalidad IPv6 a todos sus productos. (Lambrinos, 4-9 March 2007)

Network performance evaluation of VoIP on windows desktop operating systems with IPv4 and IPv6 network implementations realizado por Narayan, S. Dept. of Comput., Unitec Inst. of Technol., Auckland, New Zealand Gordon, M.; Branks, C.; Li Fan, (Narayan, Branks, & Fan, 25-27 June 2010) Donde desarrollan una investigación de rendimiento del servicio de voz IP en distintas plataformas como los son Windows Vista, Windows 7 y Windows Server 2008, también se realizan mediciones del jitter y delay para cada uno de los escenarios propuestos. Las pruebas se realizaron en dos ordenadores con hardware similares para así obtener resultados verídicos. Los resultados que obtuvieron que el rendimiento del servicio de voz depende de la versión que se utilice de IP, implementado IPv4 se obtiene un rendimiento menor que al de IPv6 en un 2% a un 6%, también se obtiene como resultado que los Codec dependen del sistema operativo que se está usando para que tengan una mejor funcionalidad. (Narayan, Branks, & Fan, 25-27 June 2010)

Interoperability between Mobile IPv4 and Mobile IPv6 based on MPLS core network realizado por Tran Cong Hung Posts & Telecommun. Technol. Inst., Ho Chi Minh Nguyen Ngoc Chan ; Nguyen Duc Thang ; Truong Dinh Huy. (Tran Cong Hung Posts & Telecommun. Technol. Inst., Thang, & Huy, 12-14 Feb. 2007) En este artículo se pretende realizar experimento con una red inalámbrica en IPv4 e IPv6, para observar la interoperabilidad entre los protocolos de IP, en este experimento también se utiliza un core de MPLS para realizar experimentos de movilidad entre los protocolos. En este artículo se obtuvo como resultado diseños de redes como base fundamental un core de MPLS ya que brinda características como velocidad de conmutación rápido y eficiente, se menciona que en estos diseños la interacción de los protocolos de IP se realiza de forma eficaz convirtiéndose en una tecnología muy utilizada ya que permite la movilidad de teléfonos cuando cambian de dominio si tener ninguna repercusión. (Tran Cong Hung Posts & Telecommun. Technol. Inst., Thang, & Huy, 12-14 Feb. 2007)

Performance Evaluation for VOIP over IP and MPLS. Realizado por Autores: Dr. Reyadh Shaker Nahum y Mohanand Maswady. En este artículo se pretende analizar el rendimiento del servicio de voz IP, se proponen dos escenarios con dos redes distintas, en el primer escenario se pretende emular una red MPLS IPv4 para realizar varias llamadas para observa a parte de jitter y el delay cuanta llamadas se pueden realizar simultáneamente con una calidad aceptable, en el segundo

escenario se pretende realizar un emulación de una red tradición IP con protocolo de enrutamiento OSPF para realizar varias llamadas para observa a parte de jitter y el delay cuanta llamadas se pueden realizar simultáneamente con una calidad aceptable. Se obtiene como resultado y conclusión que la red MPLS tiene mejor rendimiento que la red tradicional IP, ya que el en la red IP los paquetes se empiezan a perder a los 441ms y en la red MPLS a los 1200ms así mismo pasa con el jitter que en mismo tiempo empieza aumentar y el la variación del retardo o delay empieza en el mismo tiempo. Otro dato relevante que se obtiene es el número de llamadas que se pueden realizar simultáneamente en cada red, donde en la red IP tradicional se puede realizar 179 llamadas con calidad aceptable y en la red MPLS 550 llamadas con calidad aceptable. (Dr. Reyadh Shaker Nahum, 2013)

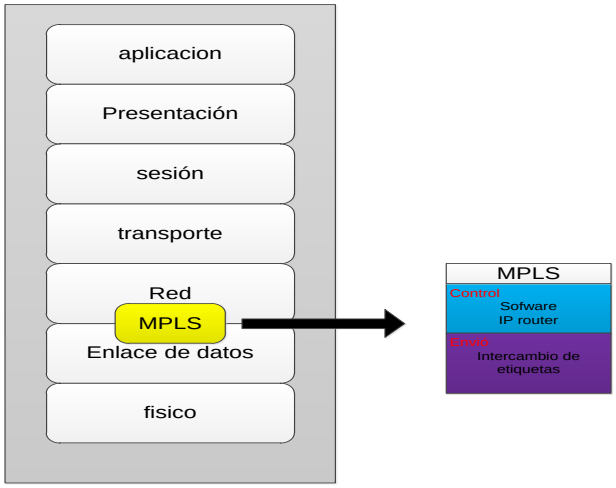
Como se mencionaba al comienzo estos artículos son lo que desarrollan temáticas más afines con el tema de este proyecto, pero ninguno cuenta con las mismas especificaciones que en este se pretender manejar, es decir los mismo sistemas como son MPLS IPv4, MPLS con método de convivencia (Tunel IPv6-Ipv4) y BGP IPv4, MPLS con método de convivencia (Tunel IPv6-Ipv4) para implementar un servicio de voz IP (IPv4-IPv6) tanto de manera emulada como de forma real, para medir el jitter y delay en cada experimento planteado.

2.2 **MPLS** (MultiprotocolLabelSwitching)

Antes de dar inicio a hablar del protocolo MPLS, se cuenta una breve historia de internet, red IP, red que en principio hacia uso de backbones compuestos por enrutadores que se conectaban entre sí. A partir del gran crecimiento del uso del internet sucedía que los operadores sufrían saturaciones en estos backbones, lo que llevó a crear una solución momentánea que consistía en aumentar el rendimiento de estos enrutadores y esto dio origen a los conmutadores ATM (Asynchronous Transfer Mode), que están orientados a la conexión, además se inicia la incorporación de un concepto importante, el QoS (calidad de servicio), donde también se integran diferentes tipos de tráfico (video, datos, voz, audio, etc.). Sin embargo, ATM presento algunos problemas, tales como la incompatibilidad de las tecnologías en capa 2 y capa 3, lo que originó protocolos que ayuden al mejoramiento del enrutamiento de paquetes, donde se busca la ruta más corta para enviar los mismos. En los últimos años se han combinado la alta velocidad de los conmutadores ATM y el enrutamiento IP, el cual da un funcionamiento óptimo. (Rekhter, 2000)

El protocolo MPLS indica conmutación de etiquetas multiprotocolo, éste está situado en el modelo OSI en medio de la capa 2 (enlace de datos) y capa 3 (red), donde MPLS toma lo más relevante de estas dos capas para optimizar el funcionamiento y se combina eficazmente el control de routing con la rapidez del switching. (Daisaku Shimazaki, october 2008)

Ilustración 2. Modelo OSI con la incorporación de MPLS.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

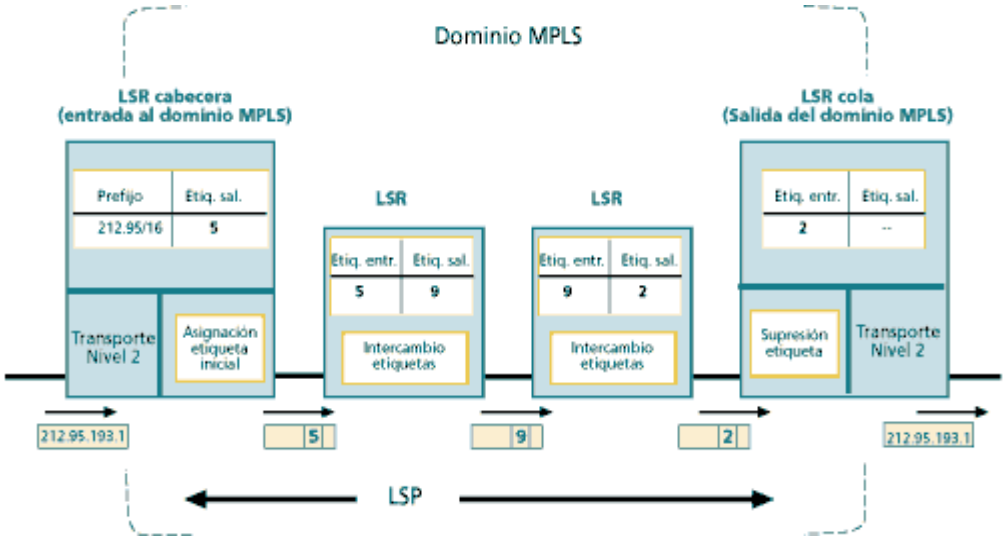
MPLS es una tecnología reciente, desarrollada por IETF (internet engineering task force), donde se buscaba la unificación de las tecnologías de transporte de datos sobre redes de circuitos y paquetes. La IETF tenía como objetivo, hablando de estándar MPLS, que esta pudiera soportar envió de paquetes unicast y multicast, que tuviera compatibilidad con todas las tecnologías de transporte no solo con ATM y a su vez debía permitir el crecimiento de internet y permitir la administración, operación y mantenimiento de las redes que ya existían. (Rekhter, 2000)

Para entender mejor el funcionamiento del estándar MPLS vamos a observar los componentes fundamentales de envió y control, los cuales constituyen la base de MPLS.

El funcionamiento de envío de paquetes se basa en la asignación e intercambio de etiquetas, lo que permite el establecimiento de los caminos LSP (label switched path), donde los caminos LSP son simples, es decir que tienen un sentido en cada punto de entrada a la red. También se puede tener un tráfico doble por lo que se necesita un LSP para cada sentido. Estos LSP son creados para unir uno o más saltos donde se intercambia la etiqueta, esto lleva a que cada paquete sea enviado

desde un Label Switchin Grouter (conmutador de etiquetas) a otro, todo esto sobre el dominio de MPLS. (Martinez, 24 noviembre del 2006)

Ilustración 3. Envío de paquete por LSP



Fuente: Evaluation of SIP Signalling and QoS for VoIP Over OLSR MANET Routing Protocol. Mazin Alshamrani, Haitham Cruickshank, Zhili Sun, Basil Elmasri, and Vahid Fami,. Cambridge : IEEE, April 2013. 978-1-4673-6421-8.

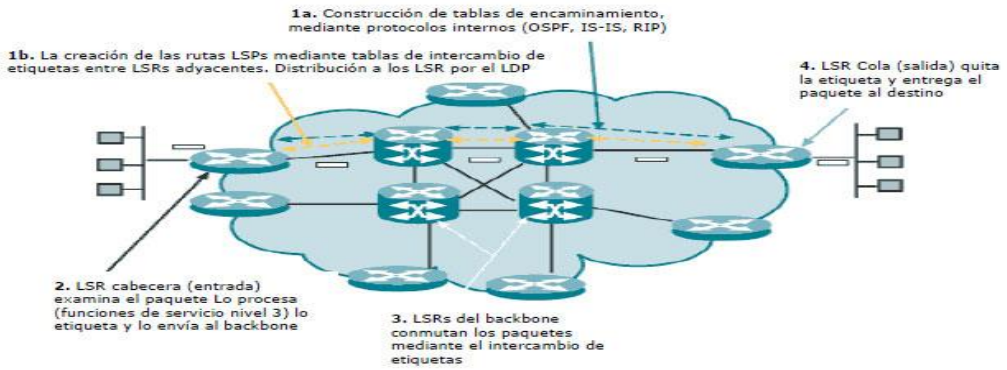
Como podemos observar en la ilustración anterior, están separados los dos componentes fundamentales de MPLS, es decir control y envío. Es importante tener en cuenta que el estándar MPLS no utiliza ninguno de los protocolos de señalización empleados por ATM. También podemos observar en la ilustración que a la entrada llega un paquete sin etiqueta el cual tiene una dirección IP de destino, el LPS revisa la tabla de encaminamiento y asigna el paquete al grupo 212.95/16, el LSR le asigna una etiqueta al paquete, después de esto la envía al siguiente LSR del LSP, teniendo en cuenta que cuando el paquete se encuentra dentro del dominio MPLS las cabeceras IP se ignoran, es decir que se guía o consulta la tabla de conmutación de etiquetas y la cambia por una nueva en cada paso que haga, cuando el paquete llega a la cola se le quita la etiqueta y hace un último salto para salir de la red MPLS y de ahí en adelante el paquete viaja por routing convencional. (Martinez, 24 noviembre del 2006)

El funcionamiento de control está compuesto por dos ítems fundamentales, el primero es como se generan las tablas de envíos de los LSP, donde en este se encuentra toda la información de la red (topología, características de los enlaces, trafico, etc.), esto lo necesita la red MPLS para poder hacer los caminos virtuales. El segundo ítem es la distribución de la información de las etiquetas a los LSR,

también conocido como información de la señalización, la cual es importante para poder establecer los circuitos virtuales, esto para la distribución de etiquetas entre los nodos. (Dirceu Cavendish, october 2004)

En el siguiente esquema se muestra el funcionamiento completo de una red MPLS en donde se puede observar que en el core (núcleo) hay una arquitectura de transporte donde los routers parece que estuvieran a un solo salto, esto hace que MPLS tenga más flexibilidad gracias a los caminos virtuales, resaltando que nunca se pierde la ubicación de los paquetes IP mientras están dentro de la red (Rekhter, 2000).

Ilustración 4. Funcionamiento de una red MPLS



Fuente: MPLS Technology and Applications. Rekhter, Bruce Davie and Yakov. san francisco : IEEE, 2000. 1-55860-656-4

Importante resaltar el objetivo fundamental que tiene la red MPLS y es la estandarización de una tecnología que intercambie etiquetas durante el proceso de reenvió de paquetes sobre el sistema actual de enrutamiento, esto con el fin de beneficiar la relación precio/desempeño. (Martinez, 24 noviembre del 2006)

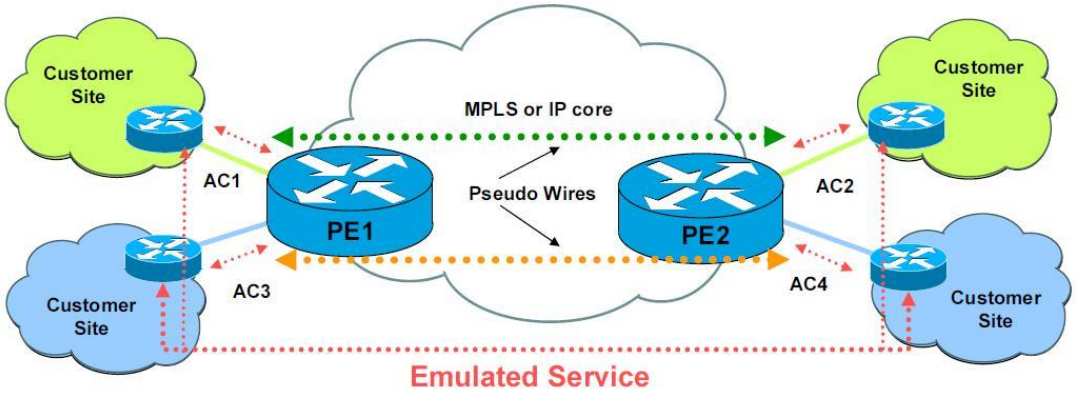
2.2.1 VPN (Virtual Private Network)

Una VPN es una red privada virtual la cual funciona sobre una red IP convencional, es decir que no cuenta con una línea dedicada, transmitiendo los datos por canales compartidos donde los paquetes pueden ser capturados pero no cifrados para ver su contenido. Tiene un comportamiento transparente para los equipos físicos que componen la red. VPN cuentas con características que la vuelven una red muy segura y fiable como que solo se pueden conectar los equipos que estén autorizados, la informa intercambiada nunca puede ser alterada y que la conexión

siempre tiene que estar disponible cuando se necesite. *(Baroncelli, Martini, & Castoldi, 2007)*

MPLS-VPN en capa 3 es denominada VRF por sus siglas en inglés (Virtual Routing and Forwarding). El funcionamiento de una VPN en capa 3 es de la siguiente manera: existen unos routers virtuales conocidos como Provider Edge (PE), son los encargados de conectar los clientes o Customer Edge (CE) a la red de transporte, existiendo un PE por cada CE que se encuentre en la red. (Baroncelli, Martini, & Castoldi, 2007) Toca tener en cuenta que el puerto de red está limitado al número de VPNS a que sean miembro y no pueden tener comunicación con dispositivos que se encuentre fuera de las VPNS a las que no esté autorizado o sea miembro. Lo anterior se puede entende mejor en la figura siguiente:

Ilustración 5. Red MPLS-VPN.



Fuente: A distributed signaling for the provisioning of on-demand VPN services in transport networks, IEEE, 2007.

Para el desarrollo de este proyecto se utilizó la VPN en los routers del fabricante Huawei, donde se implementa y es conocida como un VPN-Instance siendo la mismas VPN que se implementa en un router cisco. (Baroncelli, Martini, & Castoldi, 2007)

VRF (Virtual Routing y Forwarding): una VRF determina que política tiene cada cliente conectado al router virtual PE. La VRF cuenta con dos tablas de ruteo, una es la tabla IP y la otra es la tabla CEF o tabla de reenvió rápido de cisco, las cuales son utilizadas por las interfaces bajo un conjunto de reglas y parámetros del protocolo de ruteo que controlan la información que se incluye en l tabla de ruteo. Las VRF cuenta un papel fundamental en las VPNS ya que contienen las rutas que los clientes pueden acceder en los sitios de la VPN, se debe tener en cuenta que cada sitio puede estar suscrito a muchas VPN pero solo poder pertenecer a una

VRF. Una ventaja que tiene la VRF para evitar que se filtre o se escape tráfico de la VPN es que guarda la información de los paquetes de las tablas IP y CEF para ser reenviadas. (Baroncelli, Martini, & Castoldi, 2007)

2.3 PROTOCOLO SIP (session initiation protocol)

El protocolo de señalización SIP fue desarrollado por el grupo de trabajo MMUSIC de la entidad IETF, este protocolo tiene semejanzas a los componente más relevantes del protocolo HTTP (protocolo web) y SMTP (protocolo correo electrónico). El protocolo SIP se utiliza para la iniciación, modificación y finalización de sesión de usuarios de una red IP que utilizan servicios o aplicaciones multimedia como la voz, el video, mensajería instantánea, juegos online y realidad virtual. (Simon ZNATY, 2005) (Ming Luo, 2008)

En la actualidad la voz sobre IP (VoIP) adoptado el protocolo SIP reemplazando el protocolo que se utilizaba normalmente que era H.323, este protocolo SIP es un estándar RFC 3261. SIP fue creado con una gran simplicidad gracias a que no da detalles de la sesión si no que solo inicia, termina y modifica sesión, esto ayuda a que este protocolo se acople cómodamente a las diferentes arquitecturas y escenarios de implementación. SIP se conoce como un protocolo de petición–respuesta, donde el cliente envía una petición (requests messages) a un servidor y este servidor manda una respuesta (response messages) al cliente, teniendo en cuenta que el terminal del cliente puede realizar las dos operaciones, tanto enviar peticiones como enviar la respuesta gracias al componente conocido como agente del usuario (UAC) y servidor del agente del usuario (UAS), esto hace que pueda haber comunicaciones entre dos terminales sin que haya ningún agente intermedio. (Lu Tian1, October 2011)

El protocolo SIP tiene como obligación garantizar que la llamada realizada por el cliente llegue al destino requerido, otra característica de este protocolo es que las partes involucradas (puede ser una conferencia) ponen las pautas admitidas para este servicio. Teniendo en cuenta lo anterior, podemos decir que uno de los participantes puede gestionar ésta teniendo la capacidad de involucrar nuevos integrantes, terminar sesión de estos u otros y poderlos dejar en espera. Los usuarios pueden modificar las características durante la realización de llamada, es decir que si se estableció una llamada que solo involucre voz, pero los participantes más delante quieren otras características como vídeo lo puedan hacer. (Lu Tian1, October 2011)

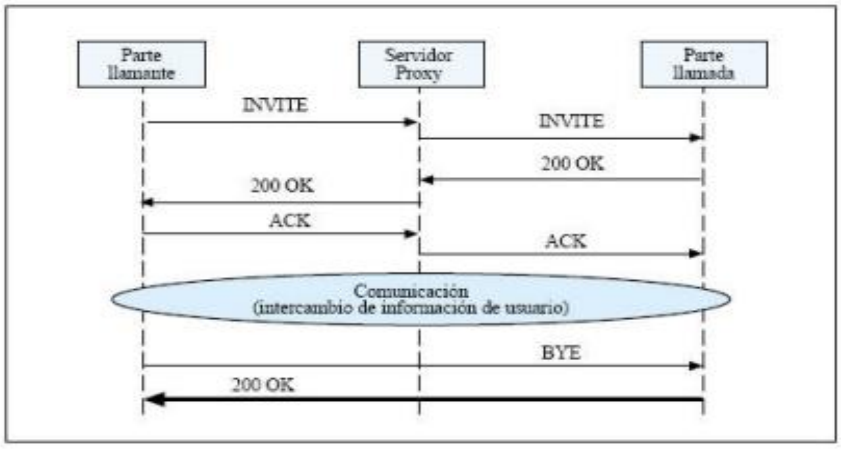
Una de las ventajas que tiene el protocolo SIP es que no tiene un modelo de direccionamiento nuevo, si no que para la creación de cuentas o usuarios se pueden

utilizar los correos electrónicos, URL y extensiones numéricas. Esto hace que el protocolo sea utilizado en gran medida para la implementación de VoIP que se utiliza en las empresas, ya que tiene diversas funcionalidades que ayudan a la comunicación simple y rápida dentro de éstas y dando beneficios extras como tener usuarios en los dispositivos móviles, los cuales se pueden comunicar con cualquier otro usuario con las mismas características: mensajería instantánea, video llamada, video conferencias entre otras. (GONZÁLEZ, 2011)

SIP está conformado por cinco componentes fundamentales para establecer la comunicación multimedia (VoIP, video llamada, mensajería instantánea, etc.):

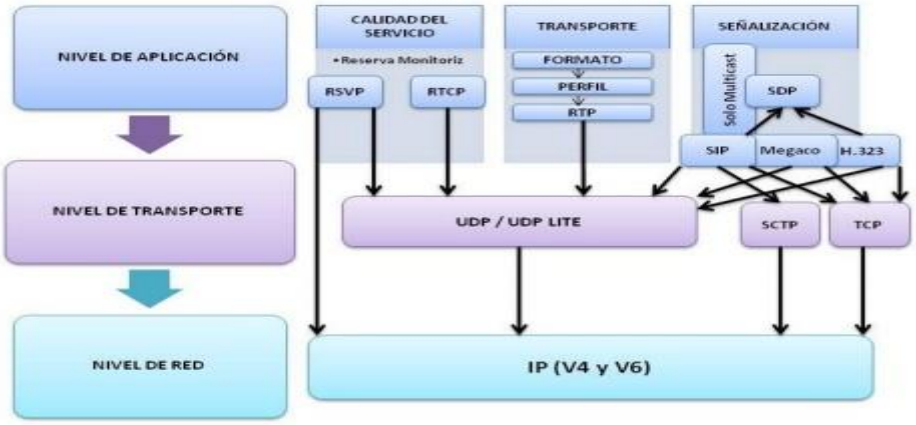
- Localización de usuarios
- Negociación de capacidades de terminales
- Disponibilidad de usuarios
- Establecimiento de llamada
- Mantenimiento de llamada

Ilustración 6. Funcionamiento del protocolo SIP



Fuente: SIP : Session Initiation Protocol. Simon ZNATY, Jean-Louis DAUPHIN y Roland GELDWERTH EFORT. s.l. : EFORT, 2005.

Ilustración 7. Protocolo sobre IP



Fuente: González, Wilmer Santamaría. Protocolos de señalización usada actualmente para terminales móviles e IP. Grupo de investigación ponente línea de investigación en telecomunicaciones. Bogotá: fundación universitaria konrad lorenz, 2011.

2.4 IPv4 E IPv6

IP es el protocolo de internet en el que está basada la red de transmisión de datos, definido por el estándar RFC 791, este estándar consiste en la transmisión de datagramas con un sistema orientado a la no conexión y no fiable, este protocolo de IP también va de la mano con el protocolo TCP, el cual se encarga de brindar las características contrarias, es decir orientado a la conexión y confiabilidad. (Óscar, 2011)

IPv4 brinda la función de best effort, quien detecta el error en la transmisión de datos, es decir que si hay pérdidas de datos los sistemas lo reconocen y da un aviso de advertencia para que estos paquetes puedan tomar otra ruta alterna. Este operación está controlada por el protocolo ICMP (internet control message protocol), el cual se encarga de enviar un mensaje al transmisor de los datos desde el punto de fallo para que vuelva a reenviar el datagrama perdido, así el equipo donde se produjo la falla puede enviar éste por otro camino y garantizar la llegada del mismo a su destino. (Mejía, 2011)

La estructura de IPv4 como se menciona anteriormente está basada en datagramas los cuales están conformados por bloques de 32 bits, donde tienen un orden especifico ya que ahí se encuentran las características de origen, destino y datos relevantes para el envió de éste. Este orden empieza por la izquierda con el bit 0 hacia la derecha con el último bit, el bloque de 32 bits se divide a su vez en cuatro octetos los cuales se expresan con un numero entre 0 a 255, lo que se traduce en un número disponible de direcciones. Otro concepto importante es la dirección de mascara de subred, la cual tiene como función identificar parte de subred de una

dirección IP. (fit, 1998) Las direcciones IPv4 se pueden clasificar en cinco grupos identificados de la siguiente manera:

Tabla 1. Clasificación direcciones IPv4

Clase	Rango	Mascas de red
A	1.0.0.0-126.255.255.255	255.0.0.0
B	128.0.0.0-191.255.255.255	255.255.0.0
C	192.0.0.0-223.255.255.255	255.255.255.0
D	224.0.0.0-239.255.255.255	255.255.255.255

Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Para entender más la clasificación de las direcciones IPv4, en la clase A se permite hasta 128 redes cada una con 16 millones de host, donde el primer octeto se usa para identificar la red y los tres siguientes para asignar los host, esta clase A gracias a la gran cantidad de host que puede tener, normalmente se utiliza en grandes compañías o en las entidades de un gobierno nacional. En la clase B se permiten 16382 redes cada una con 64000 host, donde los dos primeros octetos son para identificar la red y los dos siguiente son para asignar los host, esta clase B generalmente es usada en entornos empresariales medianos ya que cuenta con menos dirección de host para asignar pero cuenta con más redes del mismo tipo. En la clase C se cuenta con 2 millones de redes de tipo LAN cada una con 256 host, donde los tres primero octetos son para identificar la red y el último para asignar los host, esta clase son las usadas normalmente, ya que son para entornos más pequeños por su rango de direccionamiento. La clase D tiene un rango de direccionamiento que está entre la dirección 224.0.0.0 hasta la 239.255.255.255, donde la dirección IP menor es la 0.0.0.0 y la dirección más alta o mayor es la 255.255.255.255, esta clase normalmente permite multidifusión. La clase E son direcciones reservada para trabajos de investigación. (fit, 1998)

IPV6 es la nueva generación de IP, más conocido como el protocolo de internet versión 6 que fue desarrollado con el fin de ser el sucesor del protocolo anterior IPv4. Esta nueva versión del protocolo de IP cuenta con unos cambios significativos respecto a IPv4 como el incremento en el tamaño de las direcciones que pasan de 32 bits a 128 bits lo que aumenta el número de direcciones a 640 sextillones de direcciones IP. (fit, 1998) También simplificó el formato de cabecera de las direcciones para poder implementar más funciones, donde la longitud de la

cabecera es de 40 bits según establece la RFC 2460 y tiene un cambio en su estructura como lo podemos observar en la figura.8. Esta nueva versión de IP también cuenta con más seguridad en la comunicación que con la que contaba IPv4. Otro cambio considerable es la capacidad de etiqueta ya que permite el etiquetado de paquetes para la solicitud de tratamiento especial. (Mejía, 2011).

Ilustración 8. Estructura de cabecera IPv6

Fuente: **IETF.** Internet Control MessageProtocol. Marina del Rey, California: RFC 792, 1981.

Como en el protocolo de internet versión 6 (IPv6) las direcciones son demasiado extensas y se representa con números hexadecimales separados por dos puntos, donde cada número representa 16 bits, pero existe una forma de comprimir estas direcciones y es colocando dos puntos seguidor "::" en la serie de ceros que se encuentren en la dirección. En IPv6 la clasificación de las direcciones cambia totalmente a la conocida en IPv4, la clasificación va de la siguiente manera: las direcciones locales o ULA son las que se encuentran en el bloque de red FC00:: y simbolizan las redes privadas de IPv4 (fit, 1998) que están permitidas en el uso de redes privadas y pertenecen al grupo de direcciones de tipo unicast de IPv6. La segunda clasificación son direcciones anycast que pertenecen a un grupo de interfaces específicos y se encuentran representados en cualquier punto de la red, teniendo como característica que cualquier dirección de unicast puede ser también anycast. El tercer grupo de clasificación son las direcciones multicast las cuales tienen la misma función que las direcciones multicast de IPv4. (Óscar, 2011)

Algo que no se puede dejar de lado es una de las características de las direcciones IPv6 y es que no implementan direcciones broadcast.

2.5 VOZ SOBRE IP (VoIP)

La tecnología de VoIP tuvo sus inicios en el año de 1993, pero hasta el año 2004 fue desarrollada y aceptada como una tecnología. Inicialmente esta tecnología fue desarrollada para aplicaciones como Skype y Messenger donde con una conexión

de banda ancha se podían realizar llamadas gratuitas. (Ge Zhang, 2009) (Lu Tian1, October 2011) (Singh, Lamabam, & Devi, 21-22 Feb. 2014)

En los últimos años esta tecnología se ha masificado ya que permite a los usuarios tener servicios adicionales que no puede brindar una red de telefonía tradicional como son las redes telefónicas públicas conmutadas (RTPC). Esta tecnología tuvo una gran acogida en el ámbito empresarial por que brinda servicios de telefonía con un bajo costo, adicional a esto los proveedores de servicio vieron una gran posibilidad de brindar servicios novedosos. A nivel empresarial el servicio que se puede obtener con esta tecnología ayuda al crecimiento de las empresas brindando una comunicación constante y de muy buena calidad. (Ge Zhang, 2009) (Singh, Lamabam, & Devi, 21-22 Feb. 2014)

La tecnología de VoIP funciona digitalizando los paquetes de voz en paquetes de datos para ser transmitidas por redes, a la llegada a su destinos estos paquetes de datos se vuelven o través paquetes de voz, este proceso se inicia desde un teléfono que envía una señal analógica que es digitalizada en señales PCM (pulse codemodulation) por medio de codes (codificador /decodificador), la señal PCM es envía en algoritmos de compresión, donde se encuentra la voz comprimida y fraccionada en paquetes para ser transmitidas por una red WAN o LAN. Los enrutadores o Gateways pueden realizar el trabajo de codificación, decodificación y compresión dependiendo como esta configurada la red. Otro método por el cual se puede realizar el proceso de digitalización es por una PBX digital. (Taemoor Abbasi, 2005) (Singh, Lamabam, & Devi, 21-22 Feb. 2014)

Otra tecnología que es muy importante y se tiende a confundir con la VoIP es la telefonía sobre IP (ToIP), que es el conjunto de nuevas funcionalidades de telefonía que se pueden ofrecer gracias a la trasmisión de voz sobre IP en redes de datos. (Singh, Lamabam, & Devi, 21-22 Feb. 2014)

La voz sobre IP depende directamente del protocolo RTP ya que es el más utilizado para el transporte de paquetes en tiempo real, también maneja protocolos como TCP y UDP en la capa de transporte. En VoIP la señalización se puede realizar por medio de los protocolos SIP (sesión initiation protocol) adoptado como el protocolo para la señalización de VoIP, IAX, H.323, MGCP, SCCP. (Singh, Lamabam, & Devi, 21-22 Feb. 2014)

CALIDAD DE SERVICIO (QoS): La calidad de servicio es un conjunto de requisitos que debe cumplir la red para asegurar un nivel de servicio adecuado para la transmisión de datos. Otra funcionalidad de la calidad de servicio es la optimización del ancho de banda en la transmisión de tiempo real y ofrecer prioridad a

determinado tráfico dependiendo la importancia que tenga o si es en tiempo real. (Singh, Lamabam, & Devi, 21-22 Feb. 2014).

Los parámetros que se tienen en cuenta en la calidad de servicio para la VoIP son el ancho de banda, retraso temporal, variación de retardo o jitter y probabilidad de error o pérdida de paquetes. Pero estos parámetros cumplen con factores de calidad en la VoIP que son latencia o retardo, pérdida de paquetes, variación de retardo o jitter, distorsión de codificación, eco, niveles variables de señal y ruido de fondo. (Singh, Lamabam, & Devi, 21-22 Feb. 2014)

2.6 PROTOCOLO DE ENRUTAMIENTO IS-IS

Fue desarrollado originalmente por DECnet. El protocolo IS-IS (Intermediate System) permite el intercambio de información de routing entre sistemas intermedios, se puede comparar con el funcionamiento de OSPF pero eso no quiere decir que son compatibles, ambos son del tipo estado de enlace (Link State) el cual tiene como función ser proactivo, es decir tener las tablas de enrutamiento actualizadas. También tiene como característica el rechazo de las funciones de RIP, es decir puede tener trayectorias distintas de los paquetes, puede ser jerárquico, se le puede implementar QoS (calidad de servicio), soporta la autentificación, soporta una máscara de subred de longitud variable. El protocolo que permite el routing interdominio es el IDRP (Interdomain Routing Protocol) que es similar al BGP. Los IS-IS e IDRP trabajan sobre el protocolo de red CLNP. (Martinez, 24 noviembre del 2006) (Mazin Alshamrani, April 2013)

La métrica que utiliza es arbitraria y la asigna el administrador de la red, esta métrica tiene una longitud máxima de 1024 bytes. Un enlace simple puede tener un valor máximo de 64 bytes. Otra métrica adicional es el retardo del enlace. Un mapa de estos 4 tipos de métricas permite formar la QoS en el encabezado del paquete CLNP (protocolo de capa 3 en el modelo ISO) y computar la tabla de rutas de la internetwork.

2.7 PROTOCOLO DE ENRUTAMIENTO OSPF

OSPF (Open Shortest Path First) es un protocolo de enrutamiento usado dentro de un SA (Sistema Autónomo) que incorpora métricas de estado de enlace y de distancia para construir un mapa de la red y así construir el camino que va a recorrer el paquete en la red. Además, el protocolo dispone de:

- Detección de los cambios en la topología de red y restablecimiento de la ruta sin bucles.
- Baja sobrecarga, usando actualizaciones que informan de los cambios en lugar de todas las rutas.
- División de tráfico por múltiples rutas
- Encaminamiento según el tipo de servicio
- Uso de multienvio en LAN

OSPF usa el enrutamiento por áreas para generar un mapa completo del enlace en esa área, por lo tanto cuando un router pide información de la red OSPF solo da información acerca del área a la cual el router pertenece .Todos los routers con OSPF en un área mantienen una base de datos de enrutamiento que describe la topología y estado de todos los elementos en esa área. Siempre que ocurre un cambio, la información se propaga por toda el área y de esta forma el router da respuesta al problema. Un router que este iniciándose obtiene una copia de la base de datos actual de su vecino. Tras esto solo se comunican los cambios, que se llegan a conocer rápidamente pues OSPF usa un algoritmo de distribución eficiente para expandir la información de actualización por el área.

2.8 RED BGP

Es un protocolo normalizado en RFC-1771 para la versión 4, el protocolo BGP (Bourder Gateway Protocol) reemplaza al EGP en la Internet y trabaja sobre TCP. Una de las características de este protocolo es que permite tener tráfico sin loop entre sistemas autónomos, es decir entre router pares es decir IBGP para Interior o entre sistemas autónomos es decir EBGP para exterior. Normalmente es usado entre operadores ISP. La versión IBGP es más flexible, entrega varias vías de conexión en el interior del AS y dispone de una vista del exterior gracias a EBGP.

En el funcionamiento de protocolo BGP se conecta un router a la red BGP permitiendo intercambiar la tabla de direcciones completas para que haya comunicación entre los sistemas autónomos y si hay una modificación de esta tabla, la misma se intercambia de forma parcial. BGP utiliza como protocolo de transporte el TCP, una actualización de la tabla de rutas se propaga a los routers pares vecinos, es decir que se tiene una información completa de las rutas accesibles. El mapa de rutas es usado en BGP para controlar y modificar la información de routing y para definir las condiciones de redistribución de rutas entre dominios. (ITU, 2000)

Como se mencionó anteriormente BGP utiliza la información de routing intercambiadas para generar un mapa de rutas AS libre de loops. Solo un camino es conservado en la tabla de rutas y es el que se propagada a otros routers. La decisión entre distintos caminos al mismo destino se realiza mediante los siguientes atributos:

- el próximo paso (Next Hop),
- la ponderación de peso, es decir por lista de acceso o mapa de ruta y se prefiere el camino de mayor peso ponderativo
- la preferencia de ponderación local asignada por el operador donde se hace la selección de caminos con igual ponderación
- el origen de la ruta y la longitud del camino.

Un concepto utilizado en las redes BGP es routing cluster el cual tiene como función una secuencia de routers, es decir que en la mitad hay un reflector, donde ese router de la mitad refleja o repite todos los paquetes enviado por el router de origen hacia el cliente. El encabezado del paquete ocupa 19 Bytes y el mensaje tiene longitud variable. (Haitao Zheng, 2002)

2.9 PROTOCOLO RTP

El Protocolo de Transporte en Tiempo real RTP por sus siglas en inglés (Real-time Transport Protocol) este protocolo surge a raíz de la necesidad del ser humano de tener servicios en tiempo real sobre las redes existentes, uno de los servicios que más demanda tiene son las aplicaciones multimedia como los videos las música y la voz IP. (Gil Cabezas, 2009)

RTP está formado conjuntamente con el protocolo RTCP (Real-time Transport Control Protocol), es decir, el protocolo de control de transporte en tiempo real cuyo propósito es informar acerca de la calidad del transporte en la red de datos en tiempo real. (Gil Cabezas, 2009)

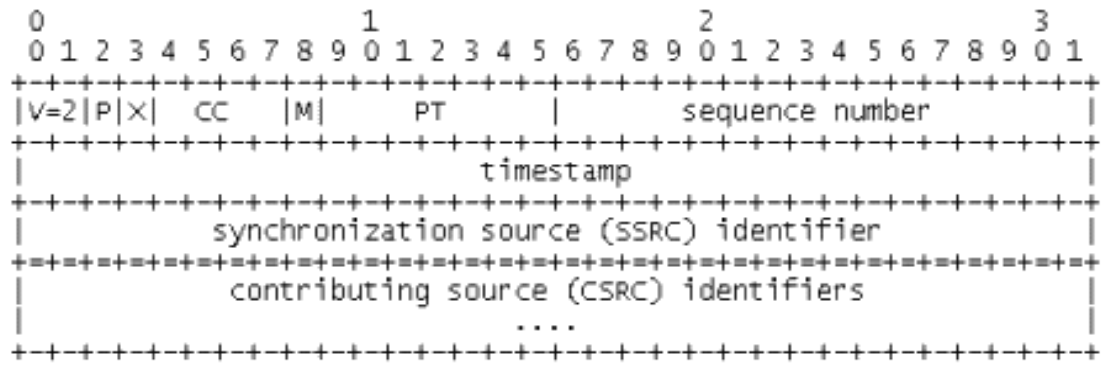
RTP está especificado en la RFC 3550.

Funcionamiento de RTP: RTP se establece generalmente sobre el protocolo de capa de transporte UDP debido a que este posee menor retardo que TCP por su condición de protocolo no orientado a la conexión, debido a esto, aunque UDP permite un flujo más rápido de paquetes, se sacrifica la confiabilidad de TCP, por lo cual RTP no garantiza la entrega de todos los paquetes; lo que indica que RTP no es un protocolo fiable. (Gil Cabezas, 2009)

RTP multiplexa varios flujos de datos en tiempo real en un solo flujo de datos de paquetes UDP, lo que genera un flujo de paquetes unicast o multicast. (16)

El encabezado RTP consiste en tres palabras de 32 bits, a continuación se muestra el formato de cabecera RTP:

Ilustración 9. IETF. RTP: A Transport Protocol for Real-Time Applications



Fuente: Network Working Group : RFC 3550, 2003.

Los primeros doce octetos están presentes en cada paquete RTP, mientras que los identificadores CSRC sólo están presentes únicamente cuando se incluyen aleatoriamente. (17)

Descripción del protocolo RTCP: El Protocolo de Control de Transporte en Tiempo Real RTCP (Real-time Transport Control Protocol) es complementario a RTP ya que le brinda a éste mecanismos de control que le permiten mejorar la calidad del flujo de datos enviados a través de la red. RTCP utiliza TCP con el fin de darle algo de confiabilidad al transporte de paquetes RTP. (Gil Cabezas, 2009)

En este proyecto se utilizara el trafico RTP, ya que los experimentos se llevaran a cabo con el servicio de voz IP sobre los sistemas MPLS y BGP, donde el protocolo RTP está directamente relacionado con el servicio de voz.

2.10 PROTOCOLO UDP

El protocolo UDP conocido así por sus siglas en inglés User Datagram Protocol que significa protocolo de datagrama de usuario, el cual proporciona una comunicación entre dos puntos de la red de manera muy sencilla. Con características como no orientado a conexión por lo que no es un protocolo estable la conexión haciendo que los paquetes se pierdan o lleguen en desorden por tal

motivo tampoco es un protocolo fiable. (Haitao Zheng, 2002) (Hongqiang Li, 27-29 June 2011)

Funcionamiento UDP: UDP utiliza el protocolo IP para transportar sus mensajes. El funcionamiento es sencillo ya que solo necesita las direcciones de origen y destino. Donde la dirección de origen es el número de puerto relacionado con la aplicación del remitente del segmento UDP. Este campo es opcional ya que si el puerto de origen no está especificado, los 16 bits de este campo se pondrán en cero. En este caso, el destinatario no podrá responder. En el puerto de destino el campo contiene el puerto correspondiente a la aplicación del equipo receptor al que se envía. Otra característica importante es la longitud ya que en este campo especifica la longitud total del segmento, con el encabezado incluido. Y por último la suma de comprobación donde la operación realizada tiene un resultado que permita controlar la integridad del segmento. (Haitao Zheng, 2002) (Hongqiang Li, 27-29 June 2011)

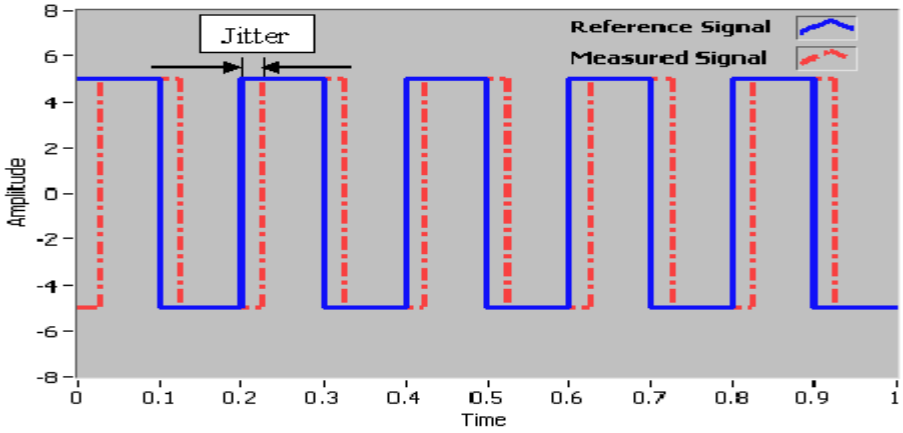
2.11 DELAY Y JITTER

El Jitter es conocido como la variación de periodo que tiene una señal digital con respecto a su posición ideal en el tiempo, la medición del jitter se realiza para observar la distorsión que tienen la señal original con respecto a la señal que llega al destino. Teniendo gran importancia en los servicios de tiempo real. El jitter entre el punto inicial y final de la comunicación debiera ser inferior a 100 ms ya que si el valor es menor a 100 ms el jitter no afecta de manera grave la transmisión de paquetes, en cambio sí es mayor toca tratarlo para poder minimizarlo. (ITU, 2000)

Posibles soluciones al jitter presentados en la transmisión de paquetes:

- Cola prioritaria: donde se asigna varios niveles de prioridad de tráfico.
- Cola definida: donde se reserva un ancho de banda para cada tipo de protocolo específico.
- Cola ponderada: mediante un algoritmo se identifica cada tipo de tráfico priorizando el de bajo ancho de banda. Esto permite estabilizar la red en los momentos de congestión.

Ilustración 10. Jitter.



Fuente: I An Introduction to Signal Generation. National Instruments. 2006.

La suma de los retardos presentados en la red es conocida como latencia, presentándose por los retardos de propagación y transmisión los cuales depende del tamaño del paquete. (ITU, 2000)

Otros modos por los que se presenta un retardo es:

- Debido a que los switch o router emiten el paquete luego de haberlo recibido y almacenado en un buffer
- Debido al proceso de reconocimiento de encabezado, errores, direcciones.

En cuanto a la medida del jitter se destacan diferentes maneras:

- Periodo del Jitter
- Ciclo periódico del jitter
- Jitter a largo plazo
- Fase de Jitter
- Intervalo de Error de Tiempo (TIE) (SITIME).

DELAY

Es conocido como en tiempo de duración del paquete desde el trasmisor hasta el receptor, afectado directamente por el medio de propagación, tiempo de ejecución de software, por la cantidad de componentes y subsistemas que conforman la red, donde cada componente tiene unas características de fábrica afectando el paquete enviado. El proceso de medida de dicho elemento se utiliza por medio de ping, este se puede utilizar para calcular la latencia de ida y vuelta. (ITU, 2000)

2.12 MECANISMOS DE CONVIVENCIA IPv4 E IPv6

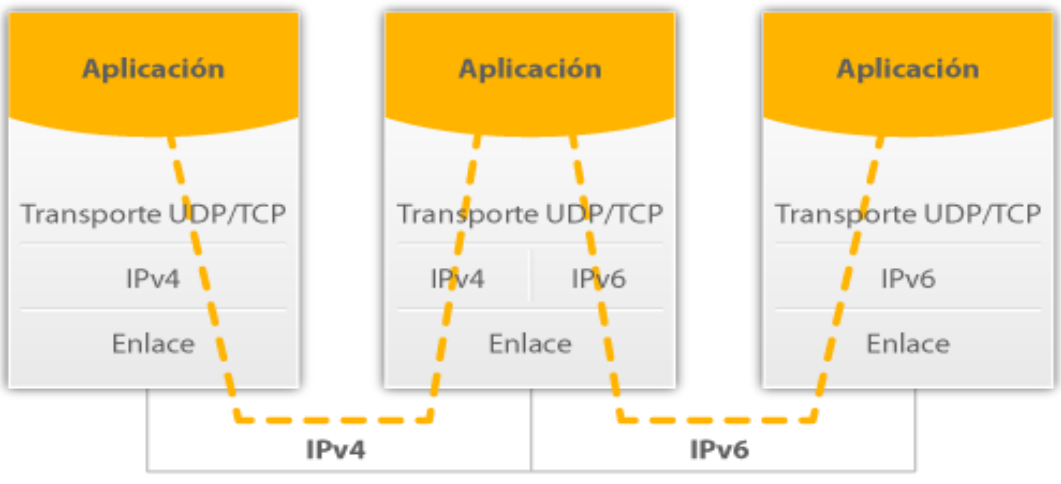
Como se conoce en la actualidad el agotamiento de las dirección IPv4 han hecho que las tecnologías empiecen a migrar al direccionamiento IPv6, buscando que esta migración no fuera de forma abrupta de una versión a otro si no que por el contrario fuera de forma progresiva. Por este motivo se buscó diseñar mecanismos de convivencia entre las dos versiones del protocolo IP. (Chen Shengyang State Key Lab. of Networking & Switching, 2010)

Con esta idea en mente los mecanismos que se implementaron para la convivencia de los dos protocolos fueron los siguientes:

- Dual stack: esta técnica funciona asignándole a los dispositivos de red direcciones IP de las dos versiones, de esta forma cuando se deseé conectar con un destino que cuente con el protocolo IPv4 utilizara la dirección IPv4 y si por el contrario el destino con el que se quiere conectar cuenta con el protocolo IPv6 utilizara la dirección en IPv6. (Chen Shengyang State Key Lab. of Networking & Switching, 2010)

  Esta técnica tiene su ventaja siendo fácil de desplegar y es ampliamente soportado, la desventaja con la que cuenta es que la topología debe contar con dos tablas de enrutamiento, teniendo que actualizar estas tablas en cada nodo para un funcionamiento eficaz. (Chen Shengyang State Key Lab. of Networking & Switching, 2010)

Ilustración 11. Técnica Dual stack.

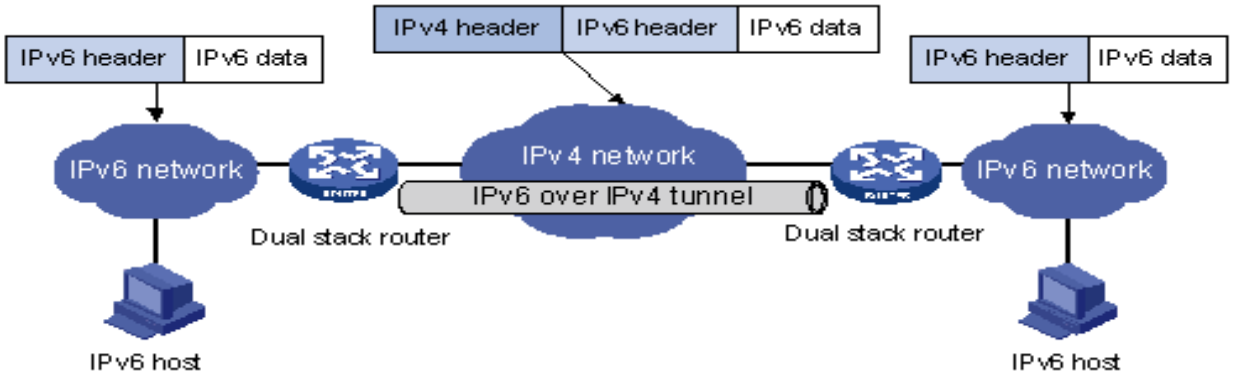


Fuente: Interoperabilidad de aplicaciones IPv4 e IPv6, Dpto. de Ingeniería y Sistemas Telemáticos Universidad Politécnica de Madrid.

- Túneles: Esta técnica fue implementada ya que todos los dispositivos que integran una red no soportaban IPv6 nativo, done este método permite conectar dispositivos que cuentan con direcciones IPv6 mediante un red en IPV4, donde se encapsulan los paquetes IPv6 en paquetes IPv4, por lo cual los paquetes IPv6 pueden ser enviados sobre la infraestructura IPv4 sin ningún inconveniente, existen gran variedad de túneles que pueden ser implementados en los que se diferencia es en la forma de encapsular los paquetes IPv6. (Chen Shengyang State Key Lab. of Networking & Switching, 2010)

  Se resalta que en este proyecto se empleó esta técnica para los experimentos en IPv6 ya que los enrutadores con lo que cuenta el laboratorio de la universidad santo tomás no soportan IPv6 en capa 2, el túnel que se implemento fue el IPv6-IPv4, este tipo de túnel crea un enlace punto a punto entre los routers dejando pasar trafico IPv6 para ser encapsulado en paquetes IPv4.

Ilustración 12. Técnica Túnel.

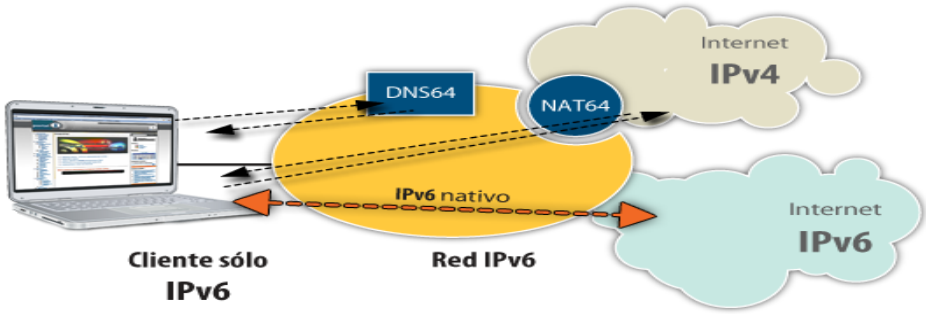


Fuente: Interoperabilidad de aplicaciones IPv4 e IPv6, Dpto. de Ingeniería y Sistemas Telemáticos Universidad Politécnica de Madrid.

- Traducción: esta técnica emplea un dispositivo traductor el cual se ubica entre las redes de IPv4 e IPv6, convirtiendo los paquetes de una protocolo a otro para poder tener comunicación. (Chen Shengyang State Key Lab. of Networking & Switching, 2010)

  Se mencionan uno de los tipos de traductor que existe: NAT-PT, TCP-UDP Relay, Socks-based Gateway, Bump-in-the-Stack, Bump-in-thea-API.

Ilustración 13. Técnica Traductor tipo NAT-PT



Fuente: Interoperabilidad de aplicaciones IPv4 e IPv6, Dpto. de Ingeniería y Sistemas Telemáticos Universidad Politécnica de Madrid.

2.13 GNS3

Es un emulador el cual permite implementar redes con características similares a las que cuenta una red real, este emulador cuenta con características que benefician a momento de configurar una red compleja como:

- Las emulaciones se realizan sin necesidad de contar con hardware como son equipos como switches, routes o computadores.

- Otra característica interesante que cuenta el software es que puede conectarse a maquinar virtuales(VitulBox, VmWare), implementando redes mucho más complejas

El funcionamiento de GNS3 es baso en el programa Dynamips siendo un emulador de IOS para enrutadores Cisco (Cisco Internetwork Operating Systems), GNS3 cuenta con los routers 1700, 2600, 3600, 3700 y 7200 para la implementación de redes soportando gran variedad de configuraciones dependiendo la versión que se tenga del emulador. (GNs3, opyright © 2007-2014 GNS3. All Rights Reserved)

2.14 Asterisk

Asterisk fue desarrollado por Mark Spencer quien en 1999 lanzó la primera central de telefonía de código abierto, la primer central Asterisk tuvo la licencia de código abierto GPL. Asterisk permite la construcción de aplicaciones de comunicación, convirtiendo un simple pc en un servidor de telefonía, por tal motivo tiene un bajo

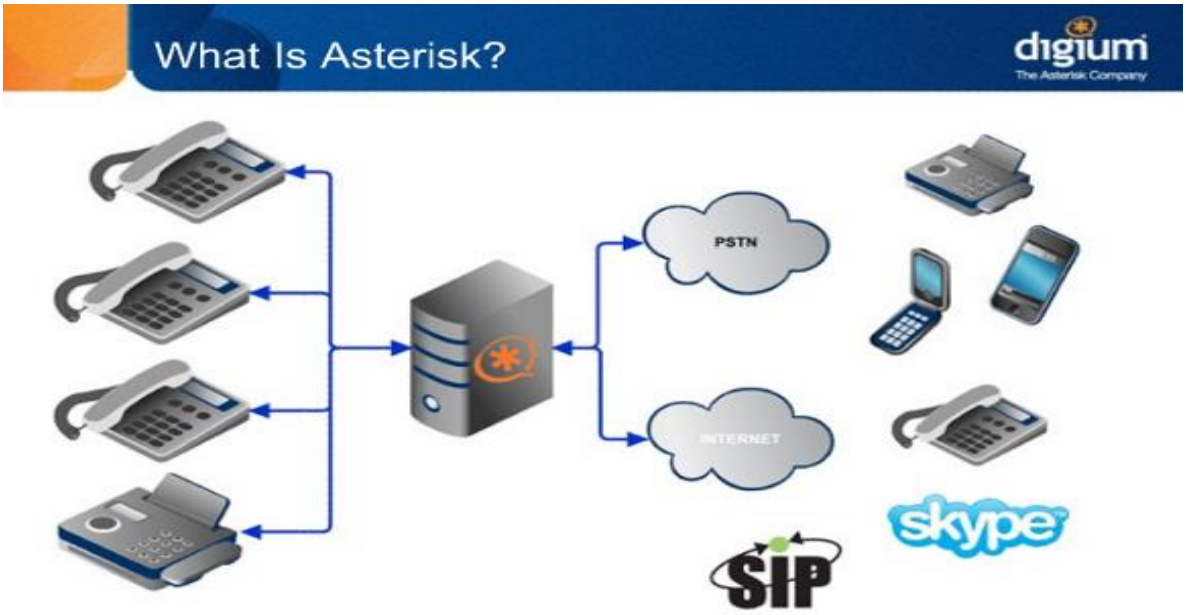
costo de implementación comprado con una central PBX tradicional. En la actualidad Asterisk se mantiene en el mercado por la compañía Digium Inc.
Una de las características con las que cuenta una central Asterisk son:

- PBX IP
- Gateway VoIP
- Correo de voz del servidor
- Puente de Conferencia
- Call Center
- IVR Servidor
- Canal SIP.

Teniendo como principal característica que convive con la red de comunicación convencional como lo es PSTN (Red Telefónica Conmutada). Con estas característica la central Asterisk es implementada en grandes, mediana y pequeñas empresas, en agencias de gobierno y muchas entidades más alrededor de todo el mundo, por los grandes beneficios con los que cuenta a un bajo costo, siendo estable y fiable. (Asterisk, 014 Digium, Inc. All Rights Reserved.)

Como se mencionó Asterisk es de código abierto dejando que las personas que utilicen esta central de telefonía puedan realizarle modificaciones para mejor el sistema y adaptarlo a sus necesidades. (Asterisk, 014 Digium, Inc. All Rights Reserved.)

Ilustración 14. Central Asterisk



Fuente: Página oficial Asterisk http://www.asterisk.org

2.15 WIRESHARK

Es una herramienta que analiza los protocolos que transitan por la red, Wireshark es la actualización de la herramienta Ethereal. El funcionamiento de esta herramienta consiste en captura y analizar los más detalladamente los paquetes que están siendo enviados por la red, identificando el protocolo al que pertenece la trama capturada. Cuenta con una interfaz gráfica fácil de manejar y con muchas funcionalidades que permite que sea unas de las mejores herramientas para el análisis de redes. Las capturas que realiza wireshark pueden ser en tiempo real o después de haberla capturado. Brinda un filtrado de información donde se puede buscar el protocolo que se necesita analizar si tener que buscar entre todos los capturados.

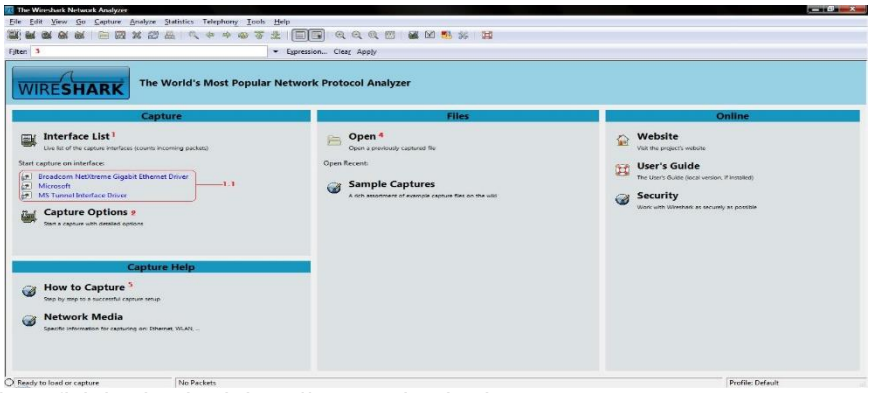
Wireshark es una herramienta gratuita y fácil de manejar, constantemente le están realizando actualización para un mejor análisis de los paquetes que pasan por una red.

Cabe resaltar que para este proyecto fue una herramienta muy importante ya que se pudo analizar el tráfico de voz que se estaba enviando por la red, capturando el protocolo RTP para poder analizar las variables a medir en este proyecto como lo es el Jitter y delay, también permitió divisar las diferentes configuración de cada sistemas utilizado (MPLS, BGP) para implementar el servicio de voz IP.

Características del analizador de redes wireshark:

- Importa y exporta paquetes en diferentes formatos
- Muestra los paquetes con información detallada
- Crea estadísticas
- Filtra información
- Abre y guarda capturas

Ilustración 15. Interfaz gráfica Wireshark



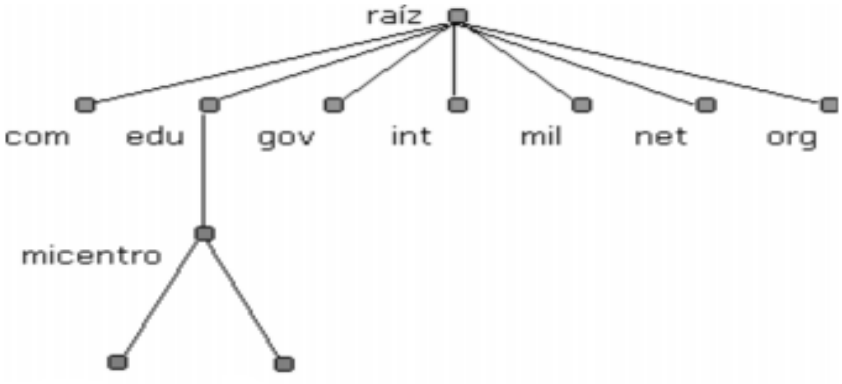
Fuente: Página oficial Wireshark http:// www.wireshark.org

2.16 SERVIDOR DNS

El servidor DNS (Domain Name System) fue implementado para facilitar la comunicación entre dispositivos en la red, ya que cada dispositivo cuenta con una dirección IP designada los cual era muy tedioso para las personas recordar las direcciones IP de los dispositivos con los que querían realizar una comunicación, por tal motivo se desarrolló un servidor que pudiera traducir las direcciones IP en nombres de domino y viceversa, es decir asignarle a una dirección IP un nombre.
La comunicación de los dispositivos se puede llevar a cabo tanto con el nombre de dominio como con la dirección IP. (Radha, Dept. of Comput. Sci. & Eng., & Selvakumar, 2011)

El servidor DNS dispone de una base de datos en la cual se almacenan todas las direcciones IP y todos los nombres de los dispositivos pertenecientes a su dominio, sin embargo no existe una base de datos única para todas las dirección y nombres de domino si no que cada servidor guarda la información correspondiente a su dominio. Los servidores DNS estas organizados jerárquicamente es decir que si el servidor más cercano no puedes responder la petición realizada este traslada la petición a un DNS superior. (Radha, Dept. of Comput. Sci. & Eng., & Selvakumar, 2011)

Ilustración 16. Jerarquización servidor DNS



Fuente: DEEPAV2: A DNS monitor tool for prevention of public IP DNS rebinding attack, Radha, B; Dept. of Comput. Sci. & Eng., Nat. Inst. of Technol., Tiruchirappalli, India ; Selvakumar, S. IEEE

En este proyecto el servidor DNS cumple un papel fundamental ya que para la implementación del servidor Asterisk IPv6, se utiliza un servidor de nombres de dominio para facilitar el registro de los teléfonos y softphone, para que se puedan registrar con el nombre de dominio Asterisk1.rumbo.edu.co.

3. DESARROLLO METODOLÓGICO DEL PROYECTO

3.1 DESCRIPCIÓN DE LA METODOLOGÍA USADA

El proyecto está enfocado al análisis del comportamiento de voz IP sobre IPv4 e IPv6, mediante experimentos para realizar mediciones del jitter y delay en sistemas MPLS y BGP. Teniendo en cuenta lo anterior, el desarrollo del proyecto se lleva a cabo en fases, comenzando por definir las variables a medir, las seleccionadas fueron el jitter y delay ya que son variables que afectan directamente servicios en tiempo real como la voz; en segundo lugar se diseñaron los experimentos a realizar enfocándose en el servicio de voz IP tanto en IPv4 como en IPv6 con el método de convivencia; en tercer lugar se plantearon los escenarios a implementar para brindar el servicio de voz, los dos escenarios a utilizar son MPLS y BGP siendo sistemas que en la actualidad tienen alta demanda en el mercado por las características con las que cuentan. Estos escenarios se pretendían implementar tanto el IPv4 nativo como en IPv6 nativo pero se presentaron problemas ya que los routers con los que cuentan los laboratorios de la Universidad Santo Tomás no soportaban VPN en capa 3 para IPv6. Dando solución a este percance implementando un túnel IPv6 sobre los sistemas MPLS IPv4 y BGP IPv4. Teniendo los escenarios definidos se organizan los experimentos buscando que, sin importar el sistema utilizado y protocolo de direccionamiento (IPv4, IPv6) todos fueran similares tanto reales como emulados. Teniendo las capturas realizadas en los experimentos con la herramienta wireshark se procede analizar y posteriormente plasmar unas recomendaciones para finalizar con unas respectivas conclusiones.

Todo el proceso mencionado anteriormente, este se desarrolló para poder resolver la pregunta planteada al comienzo del documento ¿cuál es el impacto con respecto a jitter y delay al implementar IPv4 e IPv6 en un servicio de voz en sistemas MPLS y BGP?.

3.2 VARIABLES A CONTROLAR

3.2.1. JITTER

Conociendo que un tiempo de retardo variable se define como jitter, normalmente el jitter se presenta en redes de conmutación de paquetes y no orientadas a la conexión. El jitter se define técnicamente como la variación en el tiempo en la llegada de los paquetes, causada por congestión de red, perdida de sincronización o por las diferentes rutas seguidas por los paquetes para llegar al destino. (ITU, 2000)

Debido a que el jitter afecta los servicios de tiempo real, se definió esta variable a medir ya que el servicio ha implementa en este proyecto es de voz, por lo que tiene una gran relevancia mirar que tanto afecta el servicio en los sistemas a utilizar.

3.2.2 DELAY

Es conocido como el tiempo de duración del paquete desde el trasmisor hasta el receptor, afectado directamente por el medio de propagación, tiempo de ejecución de software, por la cantidad de componentes y subsistemas que conforman la red, donde cada componente tiene unas características de fábrica afectando el paquete enviado. (ITU, 2000)

De define esta variable ya que los paquetes de voz tienen que contar con poco tiempo en la transmisión desde el transmisor hasta receptor para que no se vea afectado el servicio de forma negativa.

3.3 RECURSOS DE TOPOLOGÍA

3.3.1 Recursos Físicos

- Cuatro Routers Huawei Serie AR1220.
- Dos Routers Huawei Serie AR2220.
- Un switch Huawei Quidway® S2700 Series
- Un teléfono IP gxp-1400 grandstream
- Servidor Asterisk con sistema Operativo Ubuntu 12.04
- Un computador Lenovo con memoria RAM de 8GB, procesador Intel(R) Core(TM) i5-4200U CPU@ 1.60 GHz 2.30GHz, Sistema Operativo Windows 8 de 64 Bits.

3.3.2 Recursos De Software

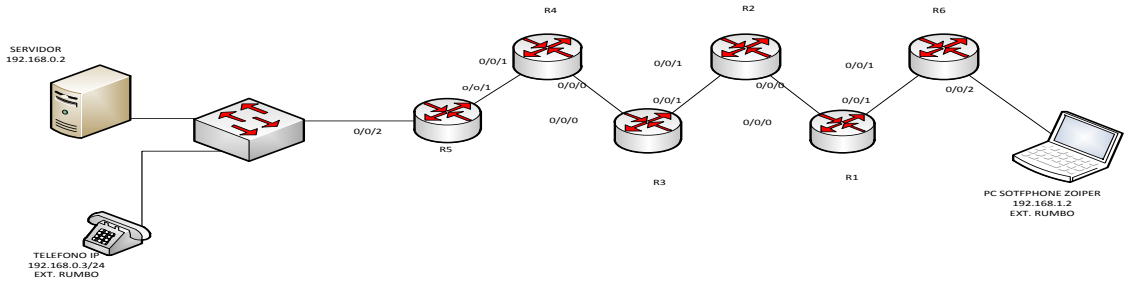
- Emulador GNS3
- Routers CISCO serie 7206VXR.
- Oracle VM VirtualBox Manager (Dos máquinas virtuales: Windows XP, Windows 7)
- Softphone Zoiper 3.6 (IPv4)
- Softphone Linphone 3.6.1 (IPv6)
- Analizador de protocolos Wireshark

3.4 DISEÑO ESCENARIOS DE PRUEBAS

3.4.1 ESCENARIOS FÍSICOS

Los escenarios se diseñaron teniendo en cuenta los equipos físicos con los que cuenta el laboratorio de la Universidad Santo Tomás. Cuatro routers AR1200 y dos routers AR2200, conectándolos de forma lineal, ubicando los routers AR2200 en los bordes cumpliendo funciones de equipos *PE* encargados de implementar las VPN y los otros cuatro Routers AR1200 en el centro de la estructura cumpliendo funciones de Routers *P*. Hacia un de los bordes se ubica un switch (Quidway S2700 Series) al que se conecta el servidor Asterisk y un teléfono IP, al otro extremo de la estructura se conecta el computador con el softphone.

Ilustración 17. Topología experimentos reales.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

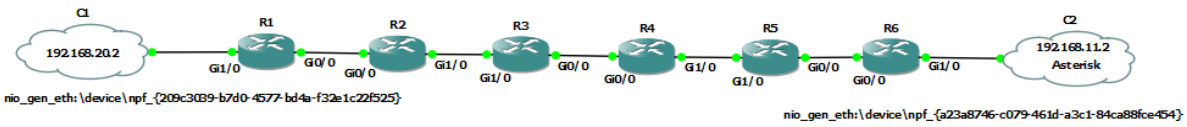
3.4.2 ESCENARIOS EMULADOS

Se desarrolla un escenario similar al físico, emulando en el software GNS3 los mismos 6 routers con los que se trabaja en el laboratorio, se conecta mediante una *nube*[1] virtual del emulador GNS3 una máquina virtual con sistema operativo Windows donde se encuentra el softphone, con otra nube se conectar el puerto

[1] La herramienta Cloud, "nube" del emulador GNS3 permite realizar conexiones entre dicho software y las interfaces de de red del sistema operativo, ya sean físicas o las correspondientes a máquinas virtuales.

Ethernet físico del computador, conectándolo a un switch (Quidway S2700 Series) al que conectamos el servidor Asterisk y un teléfono IP.

Ilustración 18. Topología experimentos emulados.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Cabe anotar que en todos los experimentos (físicos y emulados) se tendrá la misma topología, el tráfico de voz tendrá que atravesar la red y siempre tendrá las mismas características.

3.5 DISEÑO DE EXPERIMENTOS

De acuerdo con metodología del proyecto donde se realizaron pruebas sobre los escenarios diseñados midiendo y analizando las variables definidas Jitter y Delay, observando cómo afectan el servicio de voz IP sobre los dos sistemas implementados.

A continuación se muestra el diseño de los experimentos realizados en el proyecto para obtener los datos de Jitter y Delay, para posteriormente realizar el análisis de los resultados. Como se mencionó anteriormente los experimentos se diseñaron con base a los equipos con los que cuenta la Universidad Santo Tomás, los experimentos se dividen en reales y emulados. Se aclara que para todos los experimentos tantos reales como simulados se utilizó el mismo audio con duración de 25 minutos con un tamaño de 30.64MB y formato MP3, transmitido desde el altavoz de un Smartphone.

3.5.1 EXPERIMENTOS CON EQUIPOS FÍSICOS

3.5.1.1 EXPERIMENTO 1: MPLS (IPv4).

Este experimento tiene como base el MPLS en IPv4 implementando una VPN en capa 3, utilizando como protocolo de enrutamiento base OSPF, para realizar llamadas entre el teléfono IP y el softphone capturando el tráfico para medir y analizar el Jitter y Delay. En la llamada se utiliza una canción con duración de 30 minutos con un tamaño de 30.64MB.

3.5.1.2 EXPERIMENTO 2: MPLS (IPv6).

Este experimento tiene como base el protocolo MPLS en IPv4 implementando como método de convivencia IPv4-IPv6 un túnel manual que soporte trafico IPv6, ya que los equipos Huawei con los que cuenta la Universidad no soportan la VPN en capa con IPv6 nativo. Para realizar llamadas entre el teléfono IP y el softphone y posteriormente se pueda capturar el tráfico para analizar el Jitter y Delay. En la llamada se utiliza una canción con duración de 30 minutos con un tamaño de 30.64 MB.

3.5.1.3 EXPERIMENTO 3: BGP (IPv4).

Este experimento tiene como base el protocolo BGP en IPv4, para realizar llamadas entre el teléfono IP y el softphone capturando el tráfico para medir y analizar el Jitter y Delay. En la llamada se utiliza una canción con duración de 30 minutos con un tamaño de 30.64MB.

3.5.1.4 EXPERIMENTO 4: BGP (IPv6).

Este experimento tiene como base el protocolo BGP en IPv4, al igual que el experimento 2 se debió implementar un túnel manual como método de convivencia entre IPv4 e IPv6, en seguida se procederá a realizar una llamada a través de la red, con las mismas características que los experimentos anteriores para posteriormente medir Jitter y Delay.

3.5.2 EXPERIMENTOS EMULADOS

Las pruebas emuladas, se realizan con el emulador GNS3, implementando las mismas topologías de redes que se usaron en los experimentos reales. Las emulaciones se llevan a cabo en él computador Lenovo con memoria RAM de 8GB, procesador Intel(R) Core(TM) i5-4200U CPU@ 1.60 GHz 2.30GHz, Sistema Operativo Windows 8 de 64 Bits. En la llamada se utiliza una canción con duración de 30 minutos con un tamaño de 30.64 MB.

3.5.2.1 EXPERIMENTO 5: MPLS (IPv4).

Este experimento tiene como base el protocolo MPLS en IPv4 implementado en el emulador GNS3, utilizando como protocolo de enrutamiento base OSPF, para realizar llamadas entre el teléfono IP y el softphone que está ubicado en la máquina virtual, capturando el tráfico para medir y analizar el Jitter y Delay. En la llamada se utiliza una canción con duración de 30 minutos con un tamaño de 30.64MB.

3.5.2.2 EXPERIMENTO 6: MPLS (IPv6).

Este experimento tiene como base el protocolo MPLS en IPv4 implementando un túnel que soporte trafico IPv6, ya que en el proyecto se busca que todos los experimentos cuenten con la mismas características, como se mencionó anteriormente los equipos Huawei con los que cuenta la Universidad no soportan la VPN en capa con IPv6 nativo, Implementando un túnel IPv6 que conviviera con IPv4, para realizar llamadas entre el teléfono IP y el softphone que está ubicado en la máquina virtual, capturando el tráfico para medir y analizar el Jitter y Delay. En la llamada se utiliza una canción con duración de 30 minutos con un tamaño de 30.64 MB.

3.5.2.3 EXPERIMENTO 7: BGP (IPv4).

Este experimento tiene como base el protocolo BGP en IPv4, implementado en el emulador GNS3, utilizando enrutamiento mediante el protocolo OSPF, para realizar llamadas entre el teléfono IP y el softphone que está ubicado en la máquina virtual, capturando el tráfico para medir y analizar el Jitter y Delay. En la llamada se utiliza una canción con duración de 30 minutos con un tamaño de 30.64 MB.

3.5.2.4 EXPERIMENTO 8: BGP (IPv6).

Este experimento tiene como base el protocolo BGP en IPv4 implementando un túnel que soporte trafico IPv6, como se ha mencionado anteriormente el proyecto busca que tolos los escenarios cuenten con las mismas características, presentando problemas en MPLS IPv6 donde los equipos Huawei con los que cuenta la Universidad no soportan la VPN en capa 3 con IPv6 nativo, por lo que se implementó un túnel que soportara tráfico de voz IPv6 y conviviera con el protocolo IPv4, para realizar llamadas entre el teléfono IP y el softphone que está ubicado en la máquina virtual, capturando el tráfico para medir y analizar el Jitter y Delay. En la llamada se utiliza una canción con duración de 30 minutos con un tamaño de 30.64 MB.

3.6 SOFTWARE DE EMULACIÓN

Se elige el emulador GNS3 por las características con la que cuenta el programa, dejando implementar procesos OSPF, MPLS, BGP y túneles de convivencia de IPV6 con IPV4, Se compara este emulador Cisco con el emulador de Huawei llamado eNSP el cual no contaba con las mismas funcionalidades con las que cuenta el emulador Cisco ya que no soportaba el proceso MPLS siendo una de las bases para el desarrollo de este proyecto.

Tabla 2. Comparación emuladores.

GNs3	eNSP	PACKET TRACER
Interfaz gráfica de fácil manejo. Permite conectividad con otras herramientas como: • Virtual Box • Wireshark Soporta los protoclos: • RIP • OSPF • ISIS • BGP • MPLS Otras funcionalidades: • VPN • VLANS	Interfaz gráfica de fácil manejo. Permite conectividad con otras herramientas como: • Virtual Box • Wireshark Soporta los protoclos: • RIP • OSPF • ISIS • BGP Otras funcionalidades: • VPN L3 BGP • VLANS	Interfaz gráfica de fácil manejo. No permite conectividad con : • Virtual Box • Wireshark Soporta los protoclos: • OSPF • IPv6 • SSH • RST • RIP • EIGRP Otras funcionalidades: • VLANS

Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

3.7 IMPLEMENTACIÓN DE LOS ESCENARIOS

La implementación de los escenarios se lleva a cabo teniendo como base: la prestación del servicio de voz IP tanto para IPv4 como par IPv6, los sistemas MPLS y BGP. Para IPv4 los escenarios no presentan ninguna anomalía por lo que se implementan todos los procesos de forma nativa en IPv4. En IPv6 se tuvieron inconvenientes con la VPN en capa 3 ya que los equipos de la universidad no lo soportaban en IPv6 nativo, por lo que se buscó una solución viable, se implementó un túnel en todos los escenarios IPv6 para brindar el servicio de voz IPv6, y que los procesos conviviera con el protocolo IPv4 sin ningún inconveniente.

3.8 REALIZACIÓN DE EXPERIMENTOS

Los experimentos se llevaron a cabo de forma real y emulada. Para el desarrollo de los experimentos reales se ubica en uno de los extremos de la estructura un computador portátil con un softphone, al otro lado de la estructura se ubica el servidor Asterisk y un teléfono IP conectados a un switch el cual se conecta al sistema implementado (MPLS o BGP), se realiza una llamada del teléfono IP así el softphone con una duración de 25 minutos, capturando el tráfico que pasa por la red para medir y analizar el Jitter y Dealy.

4. DESARROLLO PROYECTO

4.1 DISEÑO DE EXPERIMENTOS

4.1.1 EXPERIMENTOS CON EQUIPOS FÍSICOS

4.1.1.1 EXPERIMENTO 1: MPLS (IPv4).

Este experimento tiene como base el protocolo MPLS en IPv4 implementando una VPN en capa 3, utilizando enrutamiento mediante el protocolo OSPF, realizando las mediciones de Jitter y Delay.

El direccionamiento que se utilizó en este experimento son los siguientes:

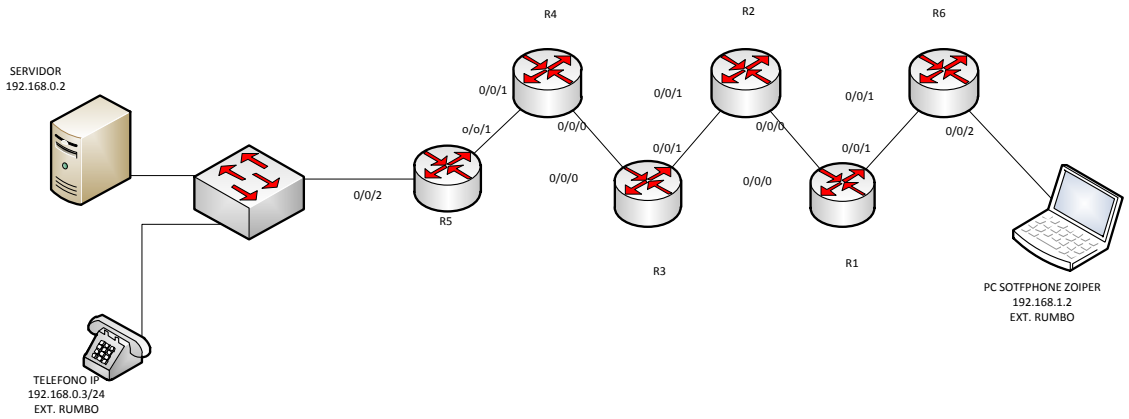
Tabla 3. Direcciones IPv4 MPLS.

	Router 1	**Router 2**	**Router 3**	**Router 4**	**Router 5**	**Router 6**
Loopback0	1.1.1.1	2.2.2.2	3.3.3.3	4.4.4.4	50.50.50.50	60.60.60.60
GE 0/0/0	10.10.12.1	10.10.12.2	10.10.34.1	10.10.34.2	N/A	N/A
GE 0/0/1	10.10.16.1	10.10.23.1	10.10.23.2	10.10.45.1	10.10.45.2	10.10.16.2
GE 0/0/2	N/A	N/A	N/A	N/A	192.168.0.1	192.168.1.1

Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Se aclara que los routers ubicados en los bordes de la estructura son el R5 y R6.

Ilustración 19. Estructura MPLS IPv4



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Para el desarrollo de este experimento se ubicaron en los extremos los routers AR2200 (R5 y R6) que cumplirán fusiones de Routers PE, los routers AR1200 se ubican al interior de la estructura en orden ascendente es decir del R1 hasta el R4. Se asigna el direccionamiento mostrado anteriormente en la tabla, en el siguiente paso se implementa el proceso OSPF, el cual realiza el enrutamiento dinámico entre todos los routers con excepción de las interfaces GE 0/0/2 de los routers R5 y R6 donde se implementó una VPN en capa tres, la cual funciona como un túnel entre las dos interfaces, por donde pasara el tráfico de voz entre el teléfono y el softphone, para realizar las pruebas de llamadas con una duración de 25 minutos y capturar el tráfico que pasa por la red para hacer los respectivos análisis.

4.1.1.2 EXPERIMENTO 2: MPLS (IPv6).

Este experimento tiene como base el protocolo MPLS en IPv4 implementando un túnel que soporte trafico IPv6, ya que los equipos Huawei con los que cuenta la Universidad no soportan la VPN en capa con IPv6 nativo, por lo que se buscó una solución viable siendo un túnel IPv6 que conviviera con IPv4. Se realizaron las pruebas de llamadas tomando mediciones de Jitter y Delay.

El direccionamiento que se utilizó en este experimento es el siguiente:

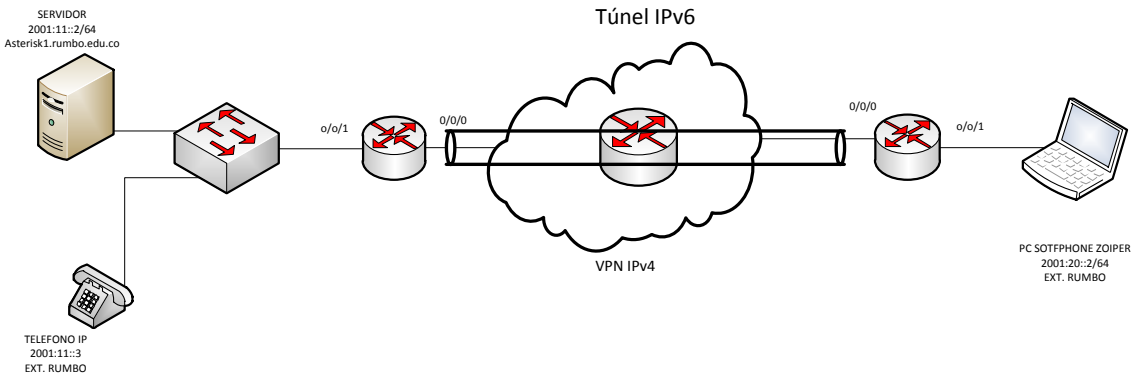
Tabla 4. Direcciones IPv6 MPLS.

	Router 1	Router 2	Router 3	Router 4	Router 5	Router 6
Loopback0	60.60.60.60	1.1.1.1	2.2.2.2	50.50.50.50	N/A	N/A
Túnel	N/A	N/A	N/A	N/A	2001:10::1	2001:10::6
GE 0/0/0	192.168.1.1	10.10.12.1	10.10.12.2	192.168.0.1	192.168.0.2	192.168.1.2
GE 0/0/1	10.10.16.2	10.10.16.1	10.10.23.1	10.10.23.2	192.168.11.1	192.168.20.1
GE 0/0/1 IPv6	N/A	N/A	N/A	N/A	2001:11::1	2001:20::1

Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Se aclara que los routers ubicados en los bordes de la estructura son el R5 y R6.

Ilustración 20. Estructura MPLS IPv6



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Para el desarrollo de este experimento se ubicaron en los extremos los routers AR2200 (R5 y R6), los routers AR1200 se ubican al interior de la estructura en orden ascendente es decir del R1 hasta el R4. Se asigna el direccionamiento mostrado anteriormente en la tabla, en el siguiente paso se implementa el proceso OSPF, el cual realiza el enrutamiento dinámico entre los routers del R1 al R4 con excepción de las interfaces GE 0/0/1 de los routers R1 y R4 donde se implementó una VPN en capa tres IPv4, en los routers R5 y R6 se implementó un túnel que soportara IPv6, el protocolo del túnel es IPv4-IPv6, donde encapsula los paquetes IPv6 en paquetes IPv4, dejando pasar el tráfico de voz IPv6 entre el teléfono y el softphone, para realizar las pruebas de llamadas con una duración de 25 minutos y capturar el tráfico que pasa por la red para hacer los respectivos análisis. Se resalta que para el experimento no se implementó la VPN en capa 2 ya que los router no contaban con esa funcionalidad.

4.1.1.3 EXPERIMENTO 3: BGP (IPv4).

Este experimento tiene como base el protocolo BGP en IPv4, utilizando enrutamiento mediante el protocolo OSPF, realizando las mediciones de Jitter y Delay.

El direccionamiento que se utilizó en este experimento es el siguiente:

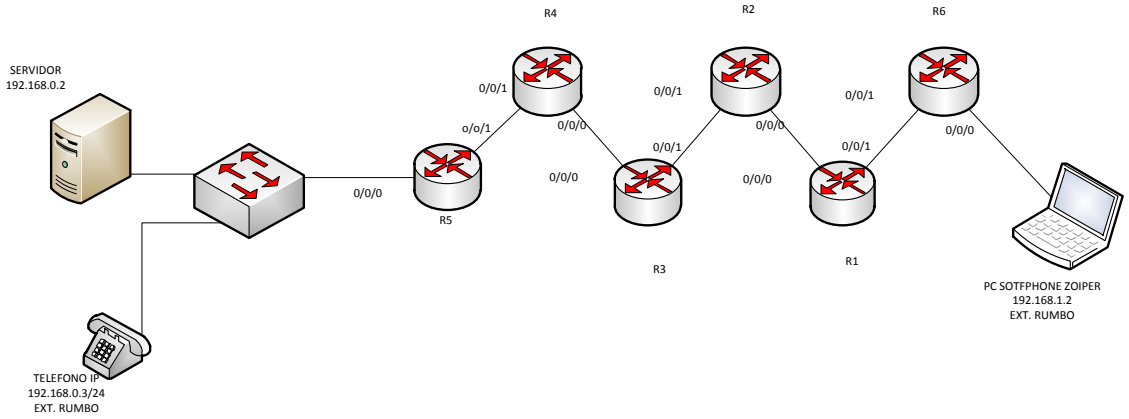
Tabla 5. Direcciones IPv4 BGP.

	Router 1	Router 2	Router 3	Router 4	Router 5	Router 6
Loopback0	1.1.1.1	2.2.2.2	3.3.3.3	4.4.4.4	50.50.50.50	60.60.60.60
GE 0/0/0	10.0.3.2	10.0.3.1	10.0.0.2	10.0.1.1	192.168.0.1	192.168.1.1
GE 0/0/1	10.0.4.1	10.0.2.2	10.0.2.1	10.0.0.2	10.0.0.1	10.0.4.2

Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Se aclara que los routers ubicados en los bordes de la estructura son el R5 y R6.

Ilustración 21. Estructura BGP IPv4



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Para el desarrollo de este experimento se ubicaron en los extremos los routers AR2200 (R5 y R6), los routers AR1200 se ubican al interior de la estructura en orden ascendente es decir del R1 hasta el R4. Se asigna el direccionamiento mostrado anteriormente en la tabla, en el siguiente paso se implementa el proceso OSPF, el cual realiza el enrutamiento dinámico entre todos los routers con excepción de las interfaces GE 0/0/2 de los routers R5 y R6 donde se implementó el protocolo BGP, intercambiando dinámicamente las direcciones que se encuentran dentro del

proceso BGP, también se indica el PEER en los routers R5 y R6 mediante la dirección de Loopback de cada uno. Dejando pasar el tráfico de voz IPv4 entre el teléfono y el softphone, para realizar las pruebas de llamadas con una duración de 25 minutos y capturar el tráfico que pasa por la red para hacer los respectivos análisis.

4.1.1.4 EXPERIMENTO 4: BGP (IPv6).

Este experimento tiene como base el protocolo BGP en IPv4 implementando un túnel que soporte trafico IPv6, ya que en este proyecto se busca que todos los experimentos realizados cuenten con las misma características, como se mencionó anteriormente se presentaron problemas en MPLS IPv6 donde los equipos Huawei con los que cuenta la Universidad no soportan la VPN en capa 3 con IPv6 nativo, por lo que se implementó un túnel que soportara tráfico de voz IPv6 y conviviera con el protocolo IPv4. Se realizaron las pruebas de llamadas tomando mediciones de Jitter y Delay.

El direccionamiento que se utilizó en este experimento es el siguiente:

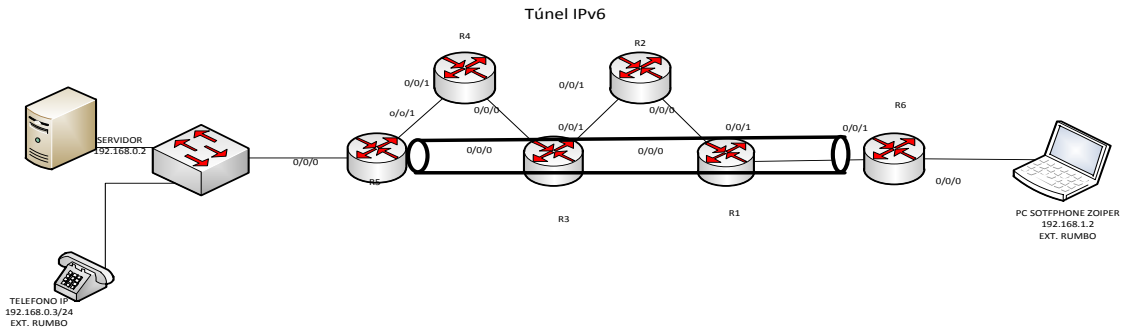
Tabla 6. Direcciones IPv6 BGP

	Router 1	Router 2	Router 3	Router 4	Router 5	Router 6
Loopback0	1.1.1.1	2.2.2.2	3.3.3.3	4.4.4.4	50.50.50.50	60.60.60.60
Túnel	N/A	N/A	N/A	N/A	2001:10::1	2001:10::6
GE 0/0/0	10.0.3.2	10.0.3.1	10.0.0.2	10.0.1.1	192.168.0.1	192.168.1.1
GE 0/0/1	10.0.4.1	10.0.2.2	10.0.2.1	10.0.0.2	10.0.0.1	10.0.4.2
GE 0/0/0 IPv6	N/A	N/A	N/A	N/A	2001:11::1	2001:20::1

Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Se aclara que los routers ubicados en los bordes de la estructura son el R5 y R6.

Ilustración 22. Estructura BGP IPv6



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Para el desarrollo de este experimento se ubicaron en los extremos los routers AR2200 (R5 y R6), los routers AR1200 se ubican al interior de la estructura en orden ascendente es decir del R1 hasta el R4. Se asigna el direccionamiento mostrado anteriormente en la tabla, en el siguiente paso se implementa el proceso OSPF, el cual realiza el enrutamiento dinámico entre todos los routers con excepción de las interfaces GE 0/0/2 de los routers R5 y R6 donde se implementó el protocolo BGP, en los routers R5 y R6 se implementó un túnel que soportara IPv6, el protocolo del túnel es IPv4-IPv6, donde encapsula los paquetes IPv6 en paquetes IPv4, dejando pasar el tráfico de voz IPv6 entre el teléfono y el softphone, para realizar las pruebas de llamadas con una duración de 25 minutos y capturar el tráfico que pasa por la red para hacer los respectivos análisis.

4.1.2 EXPERIMENTOS EMULADOS

4.1.2.1 EXPERIMENTO 5: MPLS (IPv4).

Este experimento tiene como base el protocolo MPLS en IPv4 implementado en el emulador GNS3, utilizando enrutamiento mediante el protocolo OSPF, realizando las mediciones de Jitter y Delay.

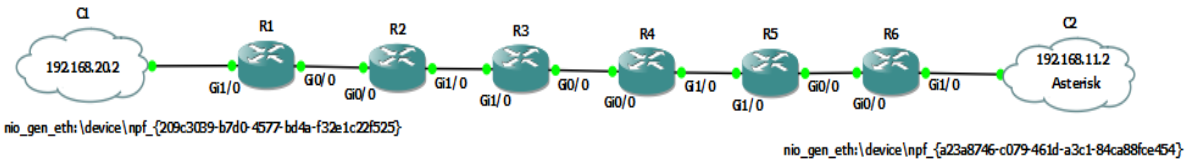
El direccionamiento que se utilizó en este experimento son los siguientes:

Tabla 7. Direcciones IPv4 MPLS.

	Router 1	**Router 2**	**Router 3**	**Router 4**	**Router 5**	**Router 6**
Loopback0	10.10.10.1	10.10.10.2	10.10.10.3	10.10.10.4	10.10.10.5	10.10.10.6
GE 0/0/0	192.168.1.2	192.168.1.3	192.168.10.1	192.168.10.2	192.168.5.1	192.168.5.2
GE 0/0/1	192.168.20.1	192.168.2.1	192.168.2.2	192.168.4.1	192.168.4.2	192.168.11.1

Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Ilustración 23. Estructura MPLS IPv4 emulada



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Para el desarrollo de este experimento se utilizan routers Cisco 7200, donde se implementa y se emula la misma topología que se usó en los experimentos reales. Se asigna el direccionamiento mostrado anteriormente en la tabla, A continuación en los routers de los bordes se conecta una nube respectivamente, en la nube de la izquierda se conecta una máquina virtual con sistema operativo Windows Xp donde se encuentra el softphone Zoiper configurado en IPv4 con la extensión 101 (USTA), en la nube de la derecha se conecta a la interfaz física del computador, la cual está conectado a un switch donde se conecta el servidor Asterisk y el teléfono IP con la extensión 103 (RUMBO). Se implementó un enrutamiento dinámico OSPF entre todos los routers con excepción de las interfaces GE 1/0 de los routers R1 y R6. Realizando las pruebas de llamadas con una duración de 25 minutos y capturar el tráfico que pasa por la red para hacer los respectivos análisis.

4.1.2.2 EXPERIMENTO 6: MPLS (IPv6).

Este experimento tiene como base el protocolo MPLS en IPv4 implementando un túnel que soporte trafico IPv6, ya que en el proyecto se busca que todos los experimentos cuenten con la mismas características, como se mencionó anteriormente los equipos Huawei con los que cuenta la Universidad no soportan la VPN en capa con IPv6 nativo, Implementando un túnel IPv6 que conviviera con IPv4. Se realizaron las pruebas de llamadas tomando mediciones de Jitter y Delay.

El direccionamiento que se utilizó en este experimento es el siguiente:

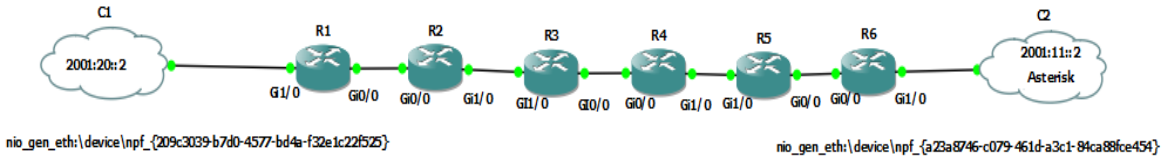
Tabla 8. Direcciones IPv6 MPLS

	Router 1	**Router 2**	**Router 3**	**Router 4**	**Router 5**	**Router 6**

Loopback0	10.10.10.1	10.10.10.2	10.10.10.3	10.10.10.4	10.10.10.5	10.10.10.6
Loopback1	2001:1::1	N/A	N/A	N/A	N/A	2001:1::6
Túnel	2001:10::1	N/A	N/A	N/A	N/A	2001:10::6
GE 0/0	192.168.1.2	192.168.1.3	192.168.10.1	192.168.10.2	192.168.5.1	192.168.5.2
GE 1/0	192.168.20.1	192.168.2.1	192.168.2.2	192.168.4.1	192.168.4.2	192.168.11.1
GE 1/0 IPv6	2001:20::1	N/A	N/A	N/A	N/A	2001:11::1

Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Ilustración 24. Estructura MPLS IPv6 emulada



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Para el desarrollo de este experimento se utilizan routers Cisco 7200, donde se implementa y se emula la misma topología que se usó en los experimentos reales. Se asigna el direccionamiento mostrado anteriormente en la tabla, A continuación en los routers de los bordes se conecta una nube respectivamente, en la nube de la izquierda se conecta una máquina virtual con sistema operativo Windows 7 donde se encuentra el softphone Linphone configurado en IPv6 con la extensión 101 (USTA), en la nube de la derecha se conecta a la interfaz física del computador, la cual está conectado a un switch donde se conecta el servidor Asterisk y el teléfono IP con la extensión 103 (RUMBO). Se implementó en los routers R1 y R6 un túnel que soportara IPv6, el protocolo del túnel es IPv4-IPv6, donde encapsula los paquetes IPv6 en paquetes IPv4, dejando pasar el tráfico de voz IPv6 entre el teléfono y el softphone, para realizar las pruebas de llamadas y capturar el tráfico que pasa por la red para hacer los respectivos análisis.

4.1.2.3 EXPERIMENTO 7: BGP (IPv4).

Este experimento tiene como base el protocolo BGP en IPv4, implementado en el emulador GNS3, utilizando enrutamiento mediante el protocolo OSPF, realizando las mediciones de Jitter y Delay.

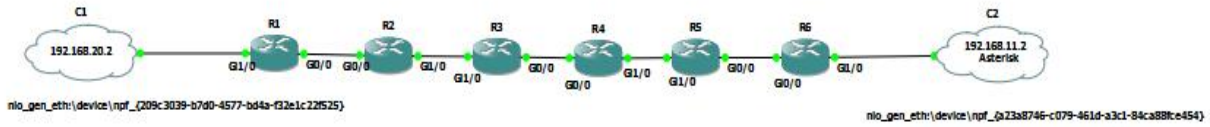
El direccionamiento que se utilizó en este experimento es el siguiente:

Tabla 9. Direcciones IPv4 BGP.

	Router 1	Router 2	Router 3	Router 4	Router 5	Router 6
Loopback0	1.1.1.1	2.2.2.2	3.3.3.3	4.4.4.4	5.5.5.5	6.6.6.6
Loopback1	11.11.11.11	22.22.22.22	33.33.33.33	44.44.44.44	55.55.55.55.55	66.66.66.66.66
GE 0/0/0	10.0.0.1	10.0.0.2	10.0.2.1	10.0.2.2	10.0.4.1	10.0.4.2
GE 0/0/1	192.168.20.1	10.0.1.1	10.0.1.2	10.0.3.1	10.0.3.2	192.168.11.1

Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Ilustración 25. Estructura BGP IPv4 emulada



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Para el desarrollo de este experimento se utilizan routers Cisco 7200, donde se implementa y se emula la misma topología que se usó en los experimentos reales para BGP. Se asigna el direccionamiento mostrado anteriormente en la tabla, A continuación en los routers de los bordes se conecta una nube respectivamente, en la nube de la Izquierda se conecta una máquina virtual con sistema operativo Windows Xp donde se encuentra el softphone Zoiper configurado en IPv4 con la extensión 101 (USTA), en la nube de la derecha se conecta a la interfaz física del computador, la cual está conectado a un switch donde se conecta el servidor Asterisk y el teléfono IP con la extensión 103 (RUMBO). Se implementó un enrutamiento dinámico OSPF entre todos los routers con excepción de las interfaces GE 1/0 de los routers R1 y R6, se configura el proceso BGP 100 para comunicar a los routers de borde y cambien información de las rutas que se encuentran al interior de la estructura. Realizando las pruebas de llamadas con una duración de 25 minutos y capturar el tráfico que pasa por la red para hacer los respectivos análisis.

4.1.2.4 EXPERIMENTO 8: BGP (IPv6).

Este experimento tiene como base el protocolo BGP en IPv4 implementando un túnel que soporte trafico IPv6, como se ha mencionado anteriormente el proyecto busca que tolos los escenarios cuenten con la mismas características, presentando

problemas en MPLS IPv6 donde los equipos Huawei con los que cuenta la Universidad no soportan la VPN en capa 3 con IPv6 nativo, por lo que se implementó un túnel que soportara tráfico de voz IPv6 y conviviera con el protocolo IPv4. Se realizaron las pruebas de llamadas tomando mediciones de Jitter y Delay.

El direccionamiento que se utilizó en este experimento es el siguiente:

Tabla 10. Direcciones IPv6 BGP.

	Router 1	Router 2	Router 3	Router 4	Router 5	Router 6
Loopback0	1.1.1.1	2.2.2.2	3.3.3.3	4.4.4.4	5.5.5.5	6.6.6.6
Loopback1	11.11.11.11	22.22.22.22	33.33.33.33	44.44.44.44	55.55.55.55.55	66.66.66.66.66
GE 0/0	10.0.0.1	10.0.0.2	10.0.2.1	10.0.2.2	10.0.4.1	10.0.4.2
GE 1/0	192.168.20.1	10.0.1.1	10.0.1.2	10.0.3.1	10.0.3.2	192.168.11.1
GE 1/0 IPv6	2001:20::1	N/A	N/A	N/A	N/A	2001:11::1
Túnel	2001:10::1	N/A	N/A	N/A	N/A	2001:10::6

Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Ilustración 26. Estructura BGP IPv6 emulada

Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Para el desarrollo de este experimento se utilizan routers Cisco 7200, donde se implementa y se emula la misma topología que se usó en los experimentos reales para BGP. Se asigna el direccionamiento mostrado anteriormente en la tabla, A continuación en los routers de los bordes se conecta una nube respectivamente, en la nube de la Izquierda se conecta una máquina virtual con sistema operativo Windows 7 donde se encuentra el softphone Linphone configurado en IPv6 con la extensión 101 (USTA), en la nube de la derecha se conecta a la interfaz física del computador, la cual está conectado a un switch donde se conecta el servidor Asterisk y el teléfono IP con la extensión 103 (RUMBO). Se implementó un

enrutamiento dinámico OSPF entre todos los routers con excepción de las interfaces GE 1/0 de los routers R1 y R6, se configura el proceso BGP 100 para comunicar a los routers de borde y cambien información de las rutas que se encuentran al interior de la estructura. En los routers R1 y R6 se implementó un túnel que soportara IPv6, el protocolo del túnel es IPv4-IPv6, donde encapsula los paquetes IPv6 en paquetes IPv4, dejando pasar el tráfico de voz IPv6 entre el teléfono y el softphone, para realizar las pruebas de llamadas con una duración de 25 minutos y capturar el tráfico que pasa por la red para hacer los respectivos análisis.

4.2 IMPLEMENTACIÓN DE LOS ESCENARIOS

La implementación se lleva a cabo por fases, siendo la primera la implementación del servicio, es decir, el servidor Asterisk, el teléfono IP, los softphone y las interfaces de red del pc. Como segunda fase encontramos la implementación de la arquitectura de red de forma física. En la tercera y última fase de implementa la arquitectura de red de manera emulada.

4.2.1 IMPLEMENTACIÓN DE SERVICIOS DE VoIP

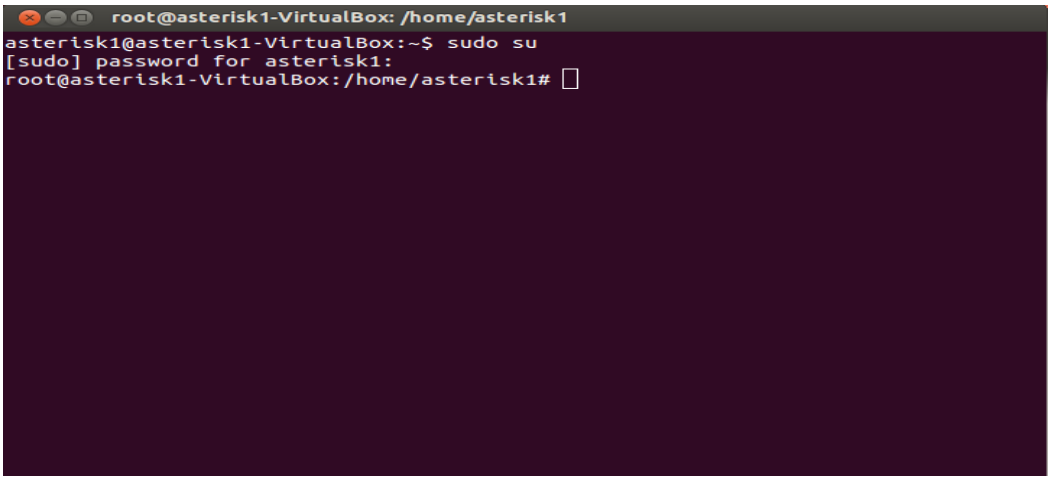
4.2.1.1 Instalación de servidores Asterisk

- **Instalación de dependencias**

Para la instalación del servidor Asterisk primero que todo se debe instalar las dependencias que ayudan al funcionamiento del servidor, antes de dar inicio con las instalaciones la consola debe estar en modo administrador o súper usuario:

Sudo su

Ilustración 27. Modo súper usuario.



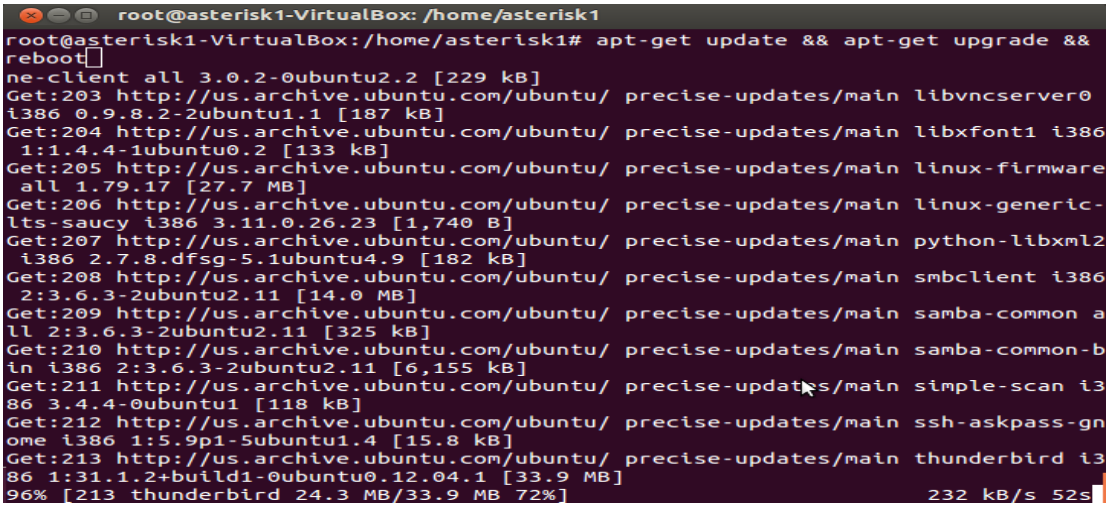
Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

En este caso el nombre de usuario del servidor es asterisk1 y la contraseña con la que se ingresa al modo administrador es rumbo2340, se debe tener en cuenta que esta contraseña es definida al momento de la instalación del sistema operativo.

Algo que también se debe tener en cuenta es que antes de instalar las dependencias con que funciona el servidor Asterisk deben instalarse las actualizaciones de las dependencias del sistema operativo y después reiniciar el servidor, esto se realiza con el siguiente comando:

apt-get update && apt-get upgrade && reboot

Ilustración 28. Instalación actualizaciones dependencias Ubuntu.

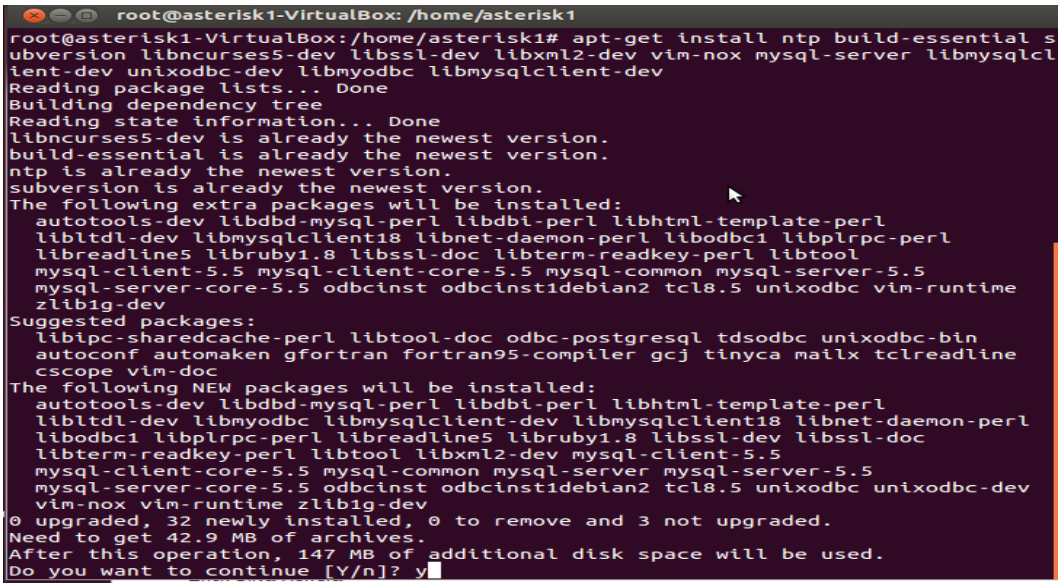


Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Con el servidor actualizado se procede a instalar las dependencias para el funcionamiento del servidor Asterisk:

apt-get install ntp build-essential subversion libncurses5-dev libssl-dev libxml2-dev vim-nox mysql-server libmysqlclient-dev unixodbc-dev libmyodbc libmysqlclient-dev

Ilustración 29. Instalación dependencias Asterisk.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Si aún no ha sido instalado MySQL en el servidor utilizado como gestor de bases de datos entonces el servidor pide establecer una contraseña para el complemento, en este caso la contraseña es la misma, rumbo2340.

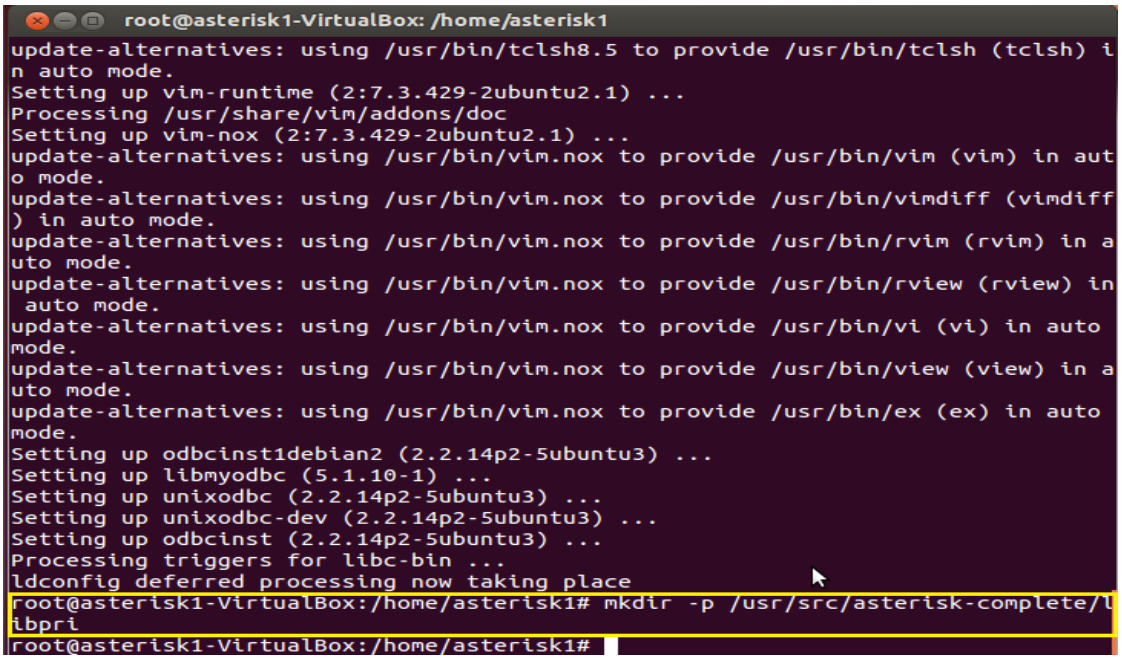
- **Instalación LibPRI**

LibPRI LibPRI es una dependencia de Asterisk y DAHDI utilizando señalización PRI para comunicar la central Asterisk con las interfaces de la red digital de servicios integrados (RDSI). Es una librería que se encarga de encapsular los protocolos utilizados para comunicarse a través de las interfaces de la red digital de servicios integrados (RDSI) como los E1.

Antes de instarla LibPRI en el servidor hay que crear la carpeta donde va estar ubicado en el servidor y desde donde dará soporte al servidor Asterisk para las conexiones:

mkdir -p /usr/src/asterisk-complete/libpri

Ilustración 30. Creación carpeta LibPRI.

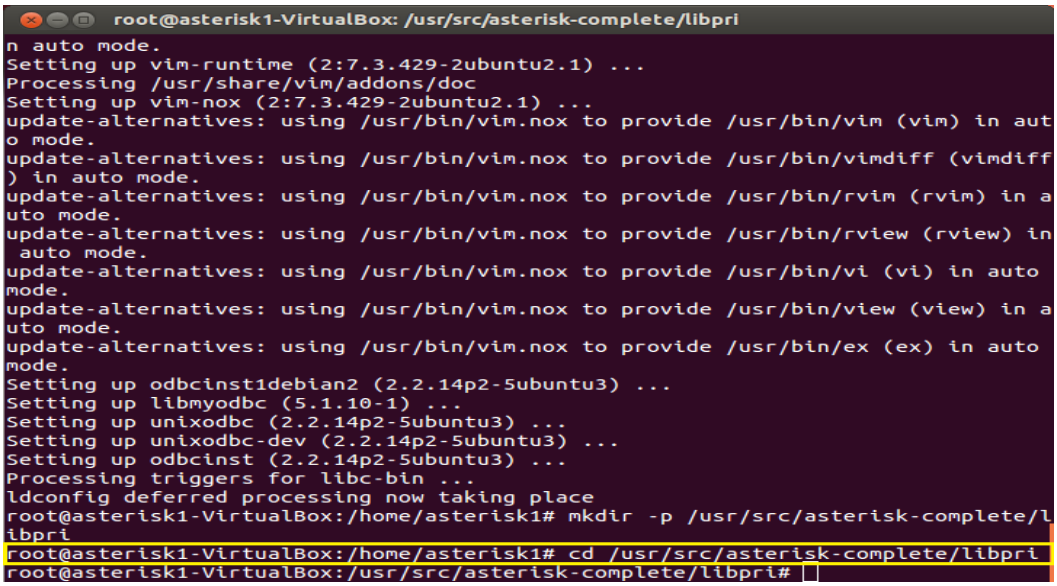


Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Luego se ingresa a la carpeta creada:

cd /usr/src/asterisk-complete/libpri

Ilustración 31. Ingreso carpeta LibPRI

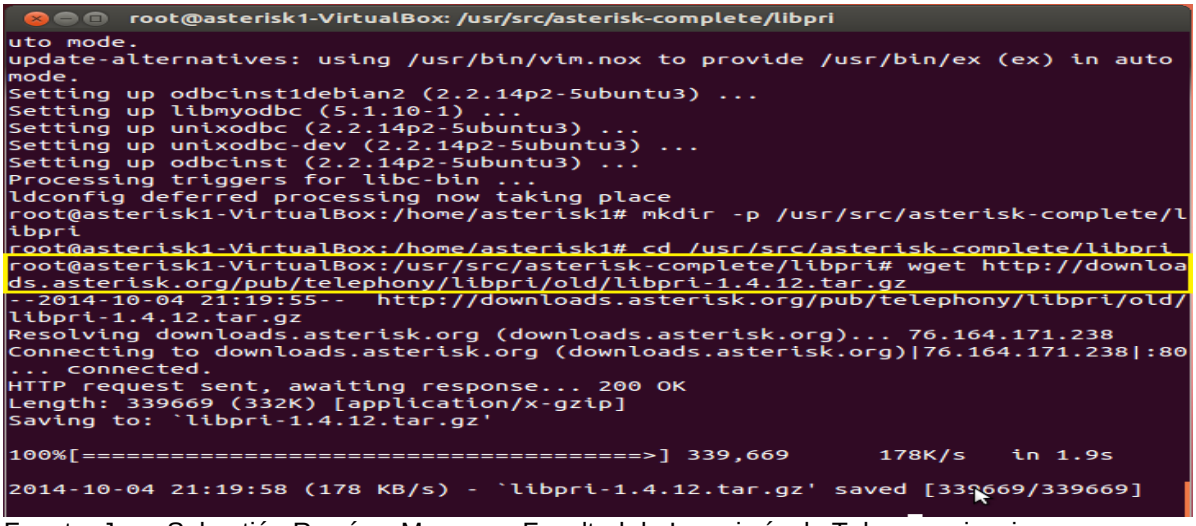


Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Después de crear la carpeta e ingresar se procede a descargar el código fuente de LibPRI desde la página oficial de Asterisk http://downloads.asterisk.org, se pone el comando con la ruta de donde se descargará LibPRI:

wget http://downloads.asterisk.org/pub/telephony/libpri/old/libpri-1.4.12.tar.gz

Ilustración 32. Descarga LibPRI.

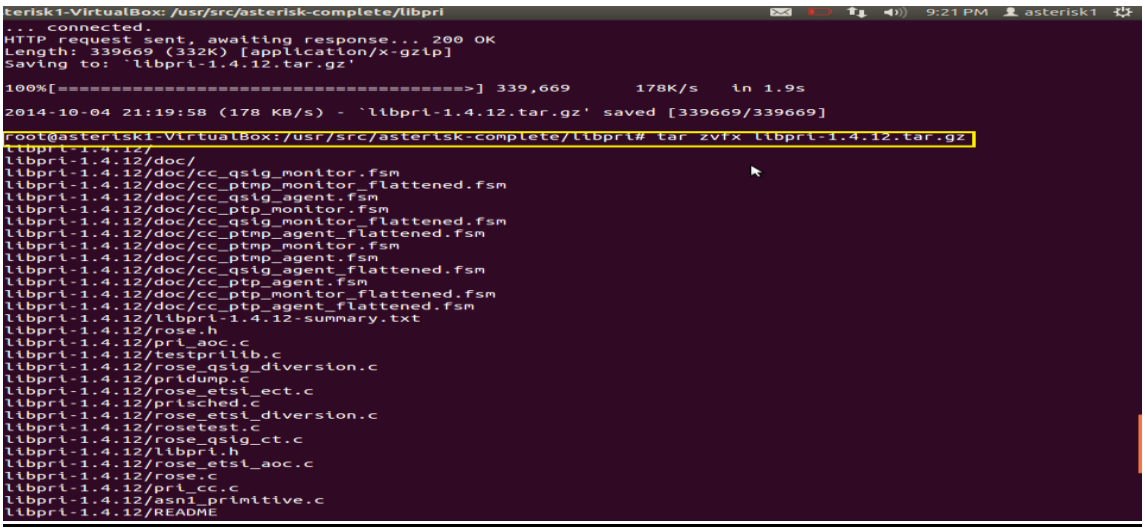


Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Teniendo el código fuente en la carpeta se procede a descomprimir el archivo tar.gz:

tar zvfx libpri-1.4.12.tar.gz

Ilustración 33. Descompresión código LibPRI

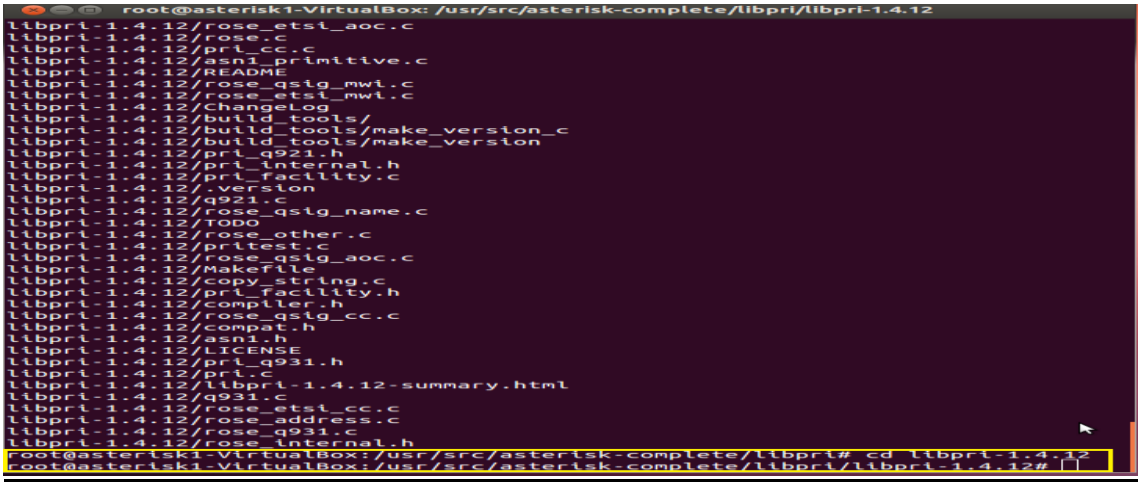


Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Descomprimiendo el archivo automáticamente se crea una carpeta donde se debe ingresar:

cd libpri-1.4.12

Ilustración 34. Ingreso carpeta descomprimida.

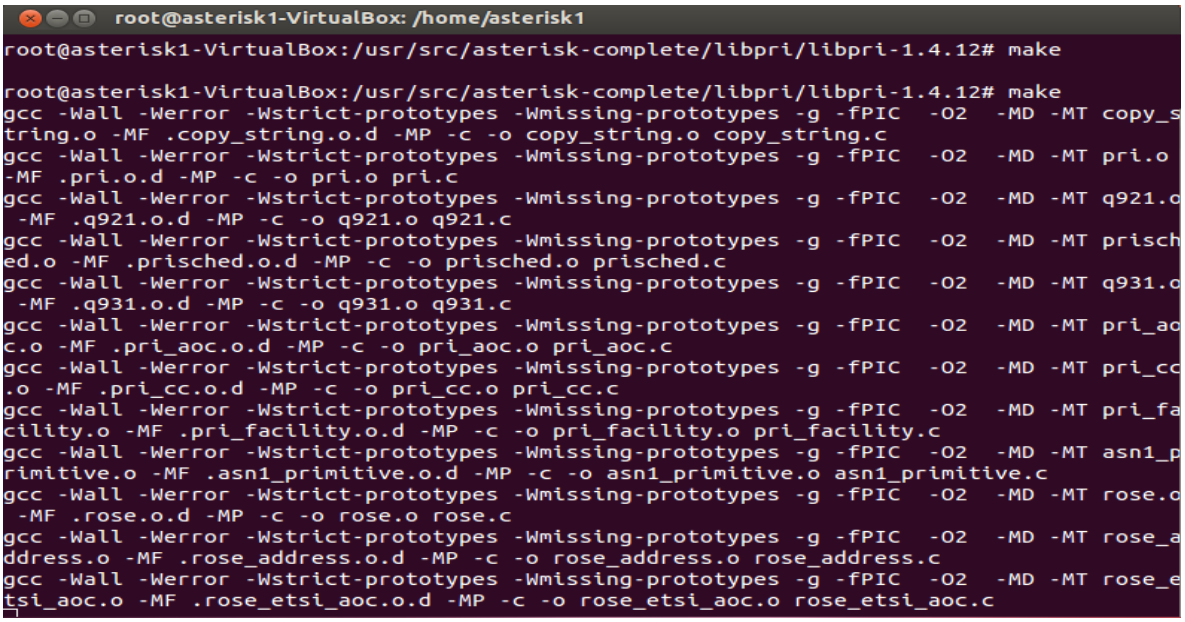


Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Y se compila el código fuente cuando se está ubicado dentro de libpri-1.4.12:

Make

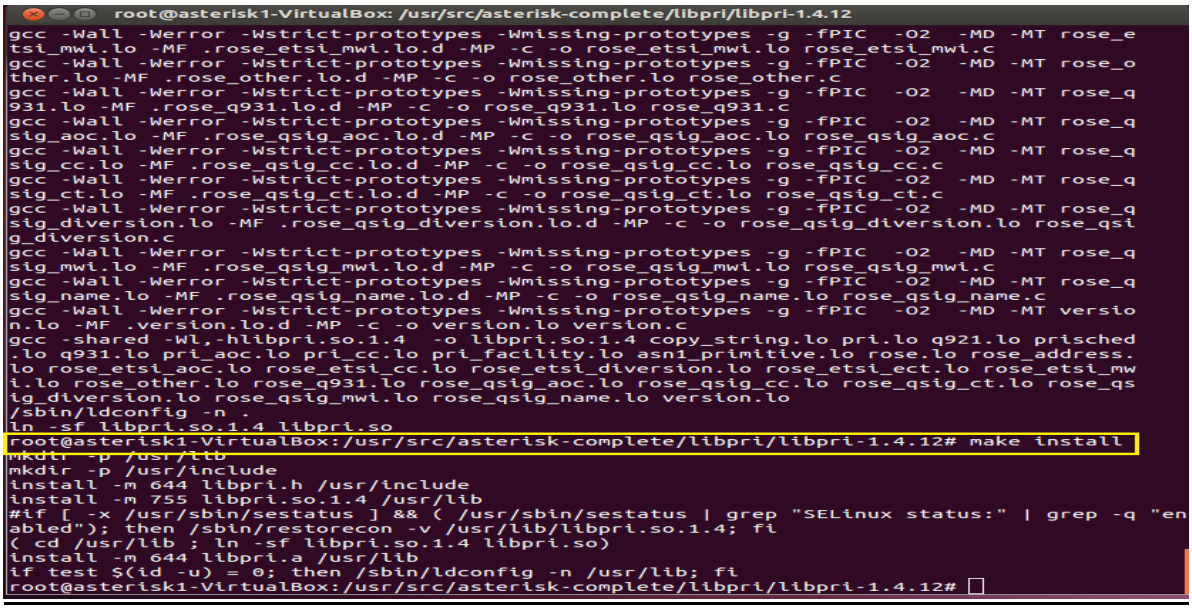
Ilustración 35. Compilación código fuente.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

make install

Ilustración 36. Compilación 2 código fuente.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

De igual manera hay que tener en cuenta que si la central Asterisk solo va ser utilizada en un entorno privado sin conexión a la red de telefonía pública conmutada no es necesario instalar LibPRI ya que como se mencionó anteriormente su objetivo es conectar el servidor Asterisk a las interfaces de la red digital de servicios integrados para tener comunicación no solo internamente.

- **Instalación DAHDI**

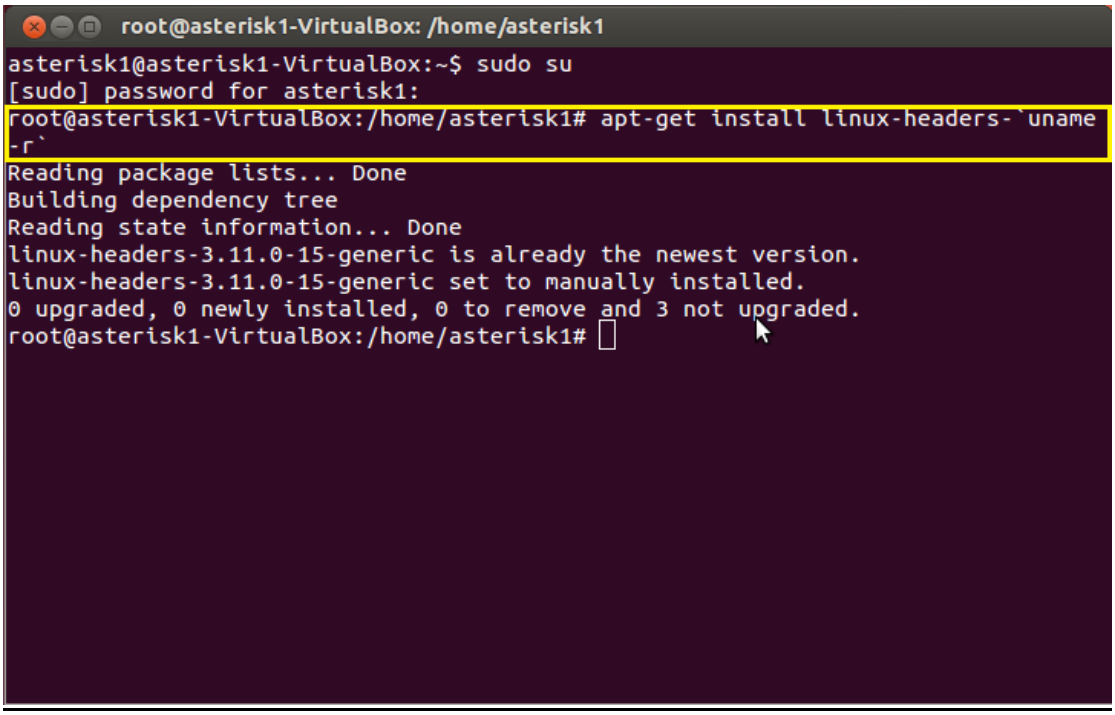
DAHDI (Digium/Asterisk Hardware Device Interface) es una interfaz que da soporte a la tarjeta y a los controladores para que el servidor Asterisk se le permita contactarse con la red telefónica pública conmutada.

La central Asterisk fue implementada en el sistema operativo Ubuntu Server 12.04 LTS, por tal motivo es necesario establecer conexiones lógicas entre el servidor y el sistema operativo, razón por la que Asterisk dispone del paquete DAHDI el cual tiene la función de proveer los controladores y la interfaz entre el hardware y el sistema operativo donde estará funcionando el sistema Asterisk.

Antes de instalar DAHDI toca instalar la dependencia Linuxheaders:

apt-get install linux-headers-`uname -r`

Ilustración 37. Instalación dependencia Linuxheraders.

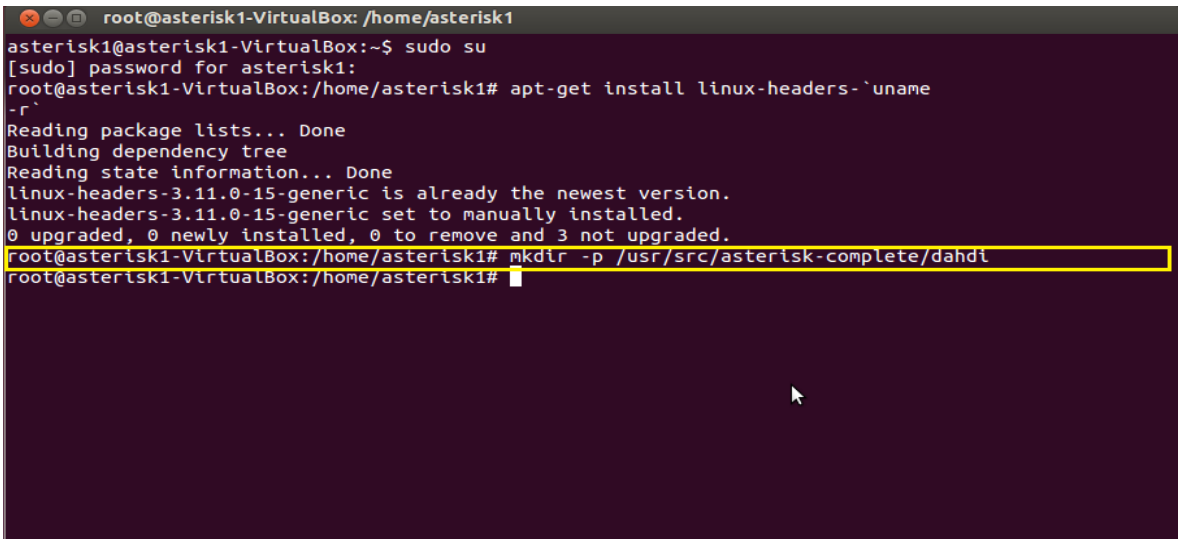


Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Ahora se crea la carpeta donde va estar ubicado el archivo con el nombre “dahdi” donde se ha instalado todo lo correspondiente a la central Asterisk:

mkdir -p /usr/src/asterisk-complete/dahdi

Ilustración 38. Creación carpeta Dahdi

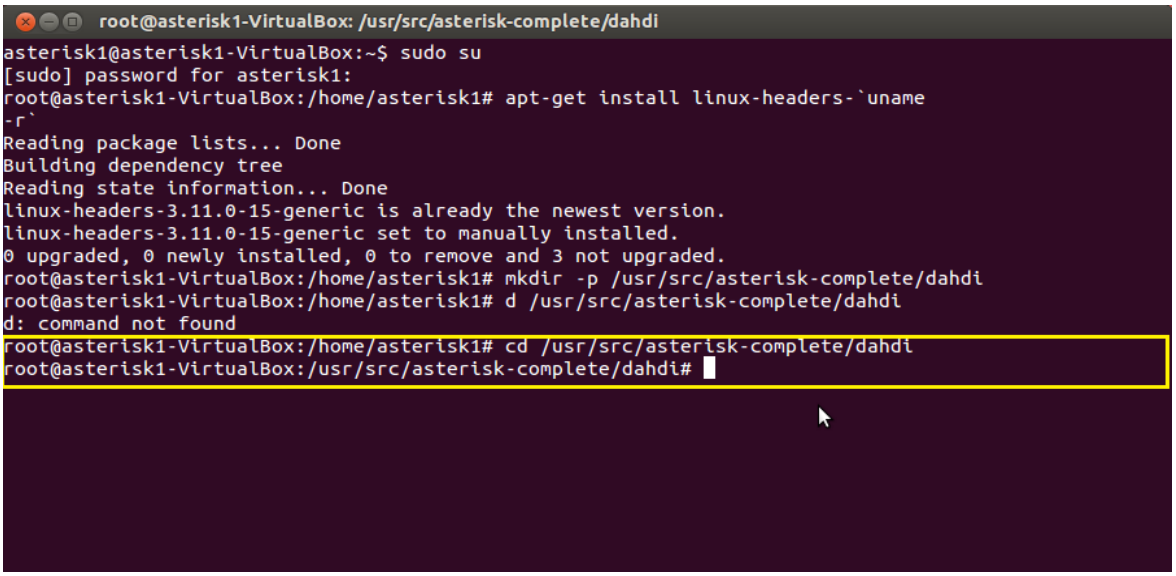


Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Se ingresa al directorio o carpeta “dahdi”:

cd /usr/src/asterisk-complete/dahdi

Ilustración 39. Ingreso carpeta Dahdi.

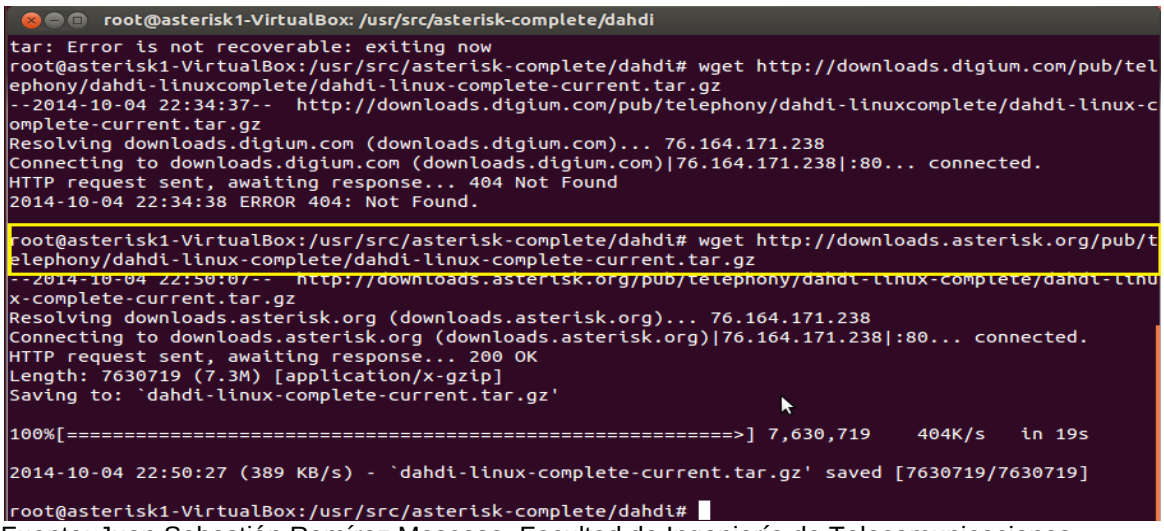


Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Se descarga el código fuente:

wget http://downloads.digium.com/pub/telephony/dahdi-linuxcomplete/dahdi-linux-complete-current.tar.gz

Ilustración 40. Descarga código fuente Dahdi.

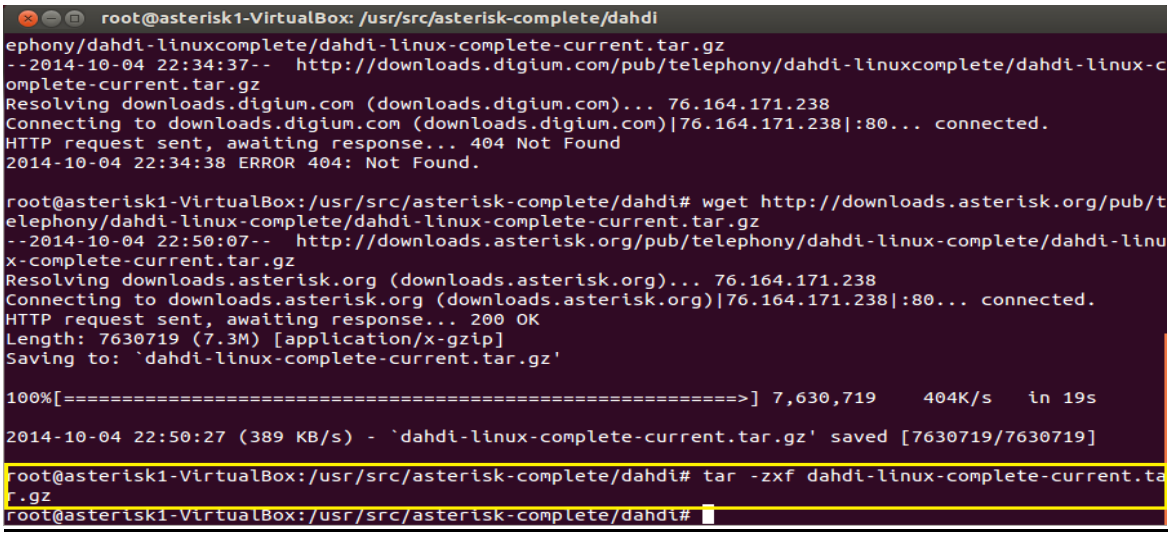


Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Y se descomprime el archivo tar.gz:

tar -zxf dahdi-linux-complete-current.tar.gz

Ilustración 41. Descomprimir Archivo Dahdi.

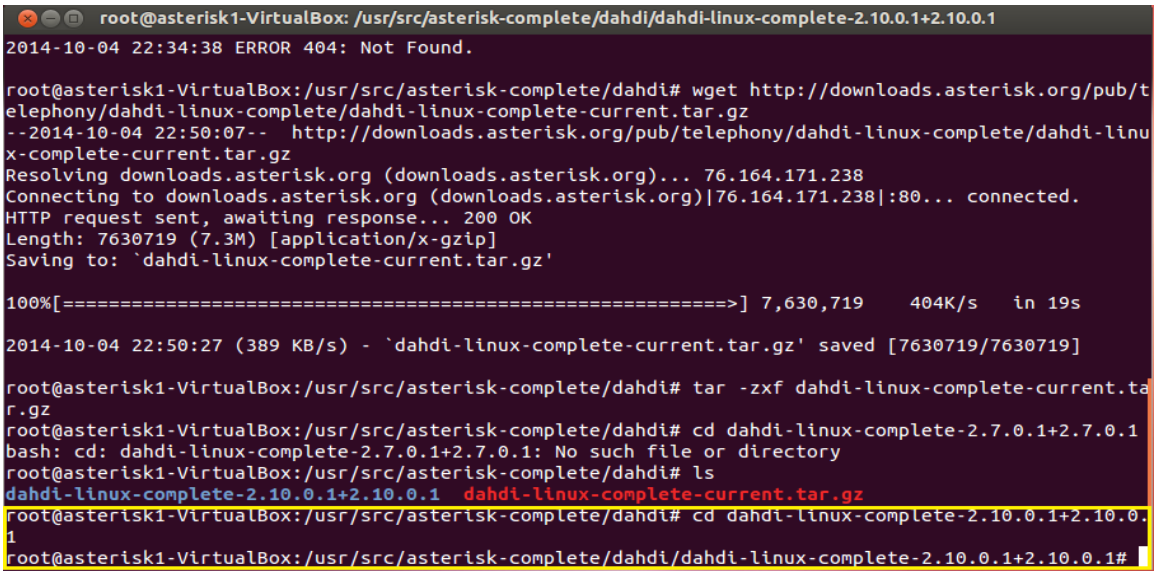


Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Descomprimiendo el archivo automáticamente se crea una carpeta donde se debe ingresar:

cd dahdi-linux-complete-2.7.0.1+2.7.0.1

Ilustración 42. Ingreso carpeta Dahdi descomprimida.

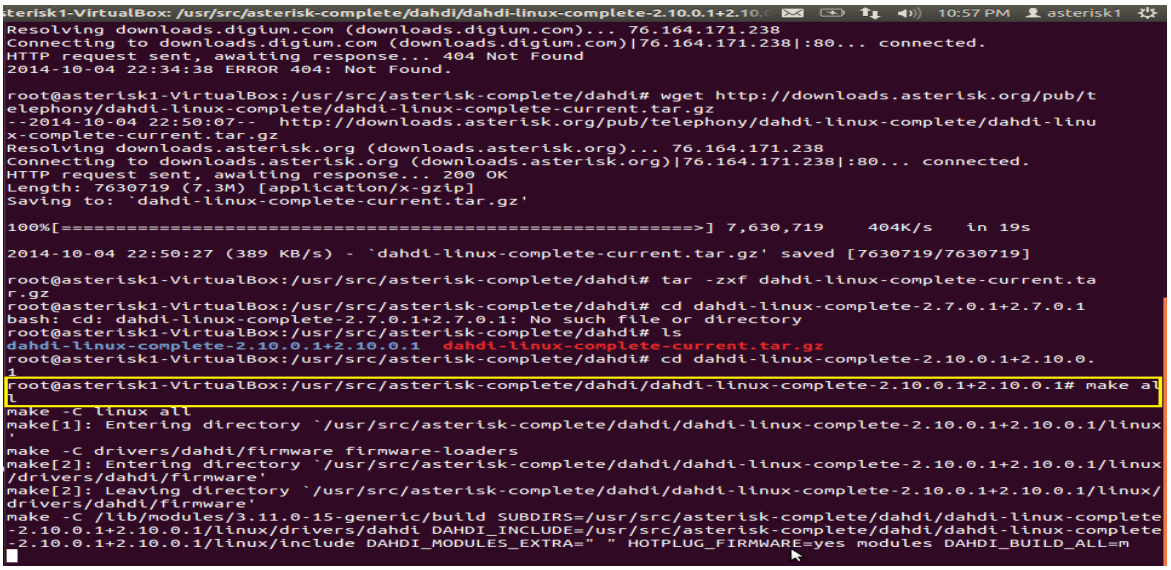


Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Se compila el código fuente:

make all

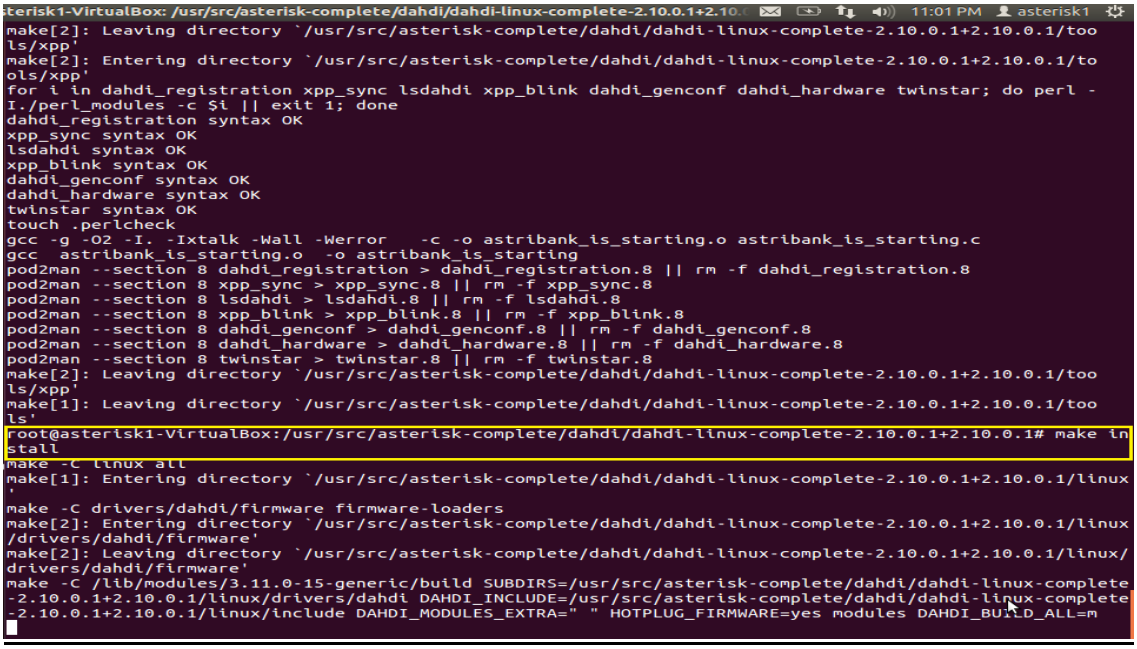
Ilustración 43. Compilar código fuente.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Make install

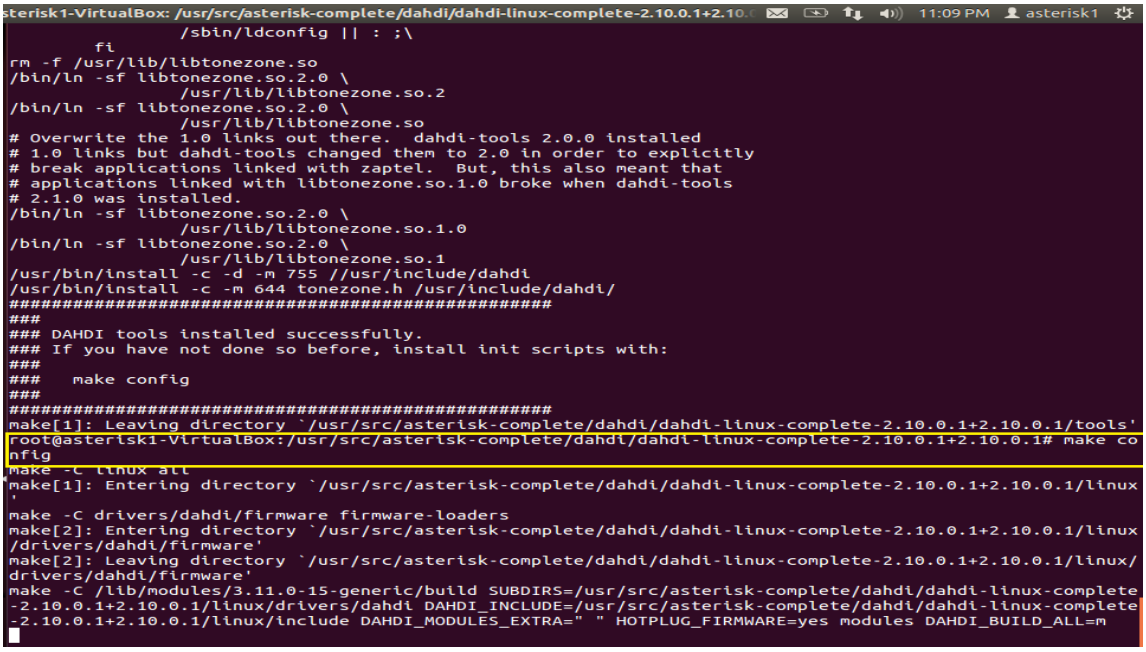
Ilustración 44. Compilar código fuente 2.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

make config

Ilustración 45. Compilar código fuente 3.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

- **Instalación de Asterisk**

Asterisk cuenta con las mismas funciones que una central de telefonía privada PBX brindando servicios como de VoIP corporativos, conexiones a líneas de telefonía análoga y conexiones a la RDSI o a la RTPC para servicios de voz, pero con la diferencia más sobresaliente que es un software de versión libre por lo cual los costos son demasiado bajos a los utilizados para implementar una PBX. Asterisk está programada en lenguaje C y puede ser compilado sobre servidores con sistemas operativos basados en GNU/Linux.

Como se mencionó anteriormente la implementación de una central Asterisk es mucho más económica que la implementación de una central PBX, ya que solo requiere un equipo con un procesamiento relativamente bajo, (dependiendo del entorno de red) y la adecuada programación de servicios de voz como extensiones, IVR (Interactive Voice Response), buzón de voz, troncales, Gateways a las redes de voz clásicas y todo tipo de servicios de voz que prestan las costosas PBX.

Una ventaja con la que cuenta la central Asterisk es que sus versiones son continuamente actualizadas agregándole más servicios y corrigiendo las falencias con las que cuenta, para este proyecto se utiliza la versión 1.8, ya que fue la primera

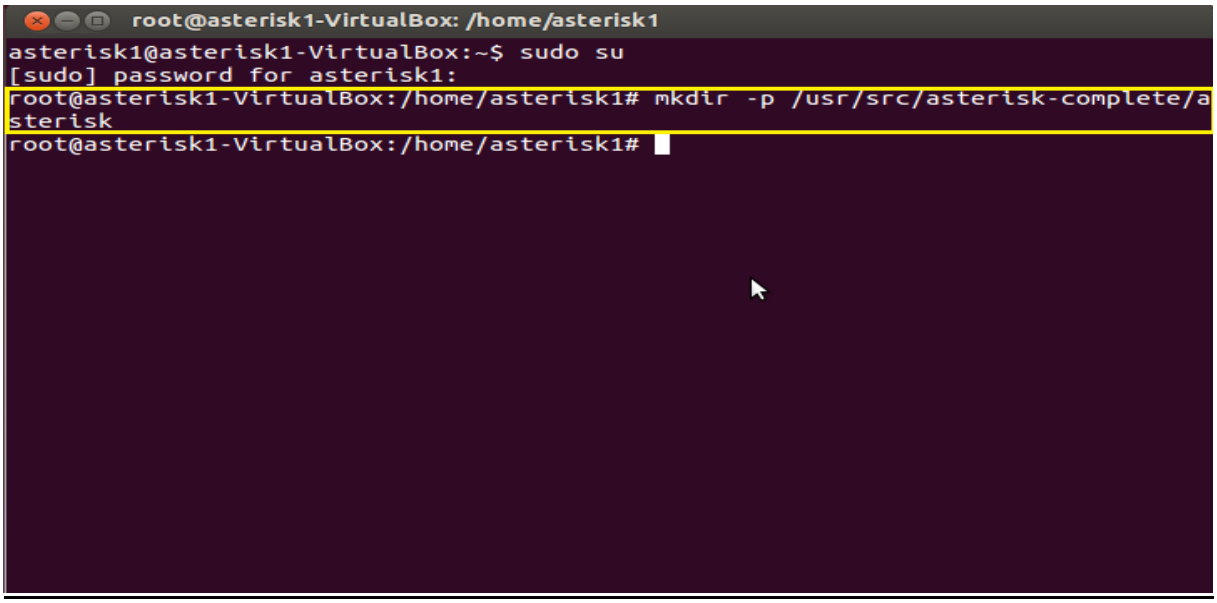
que cuenta con la característica de implementar el servicio de voz IP con el protocolo IPv6.

Una capacidad que tiene la central Asterisk es que reconoce los protocolos como H.323, SIP MGCP y IAX.

Ahora se crea la carpeta donde va estar ubicado el archivo con el nombre “asterisk-complete” donde se instalara el modulo que contiene al sistemas Asterisk:

mkdir -p /usr/src/asterisk-complete/asterisk

Ilustración 46. Creación carpeta ubicación módulo Asterisk.

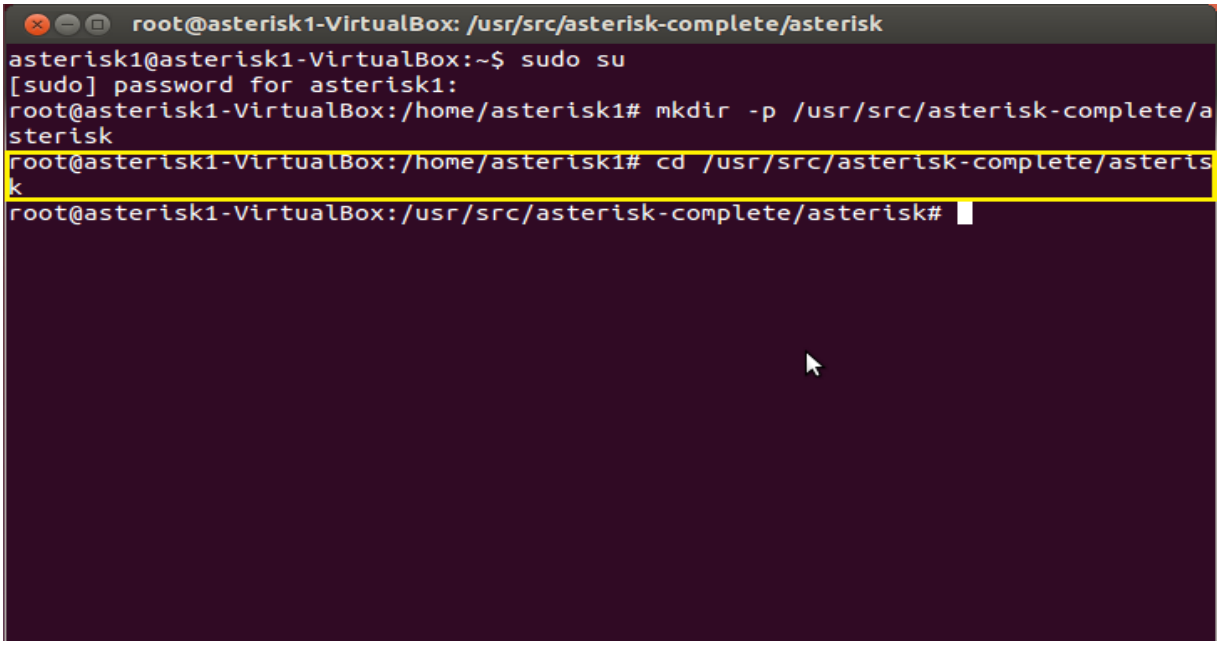


Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Luego se ingresa al directorio “asterisk”:

cd /usr/src/asterisk-complete/asterisk

Ilustración 47. Ingresar carpeta Asterisk.

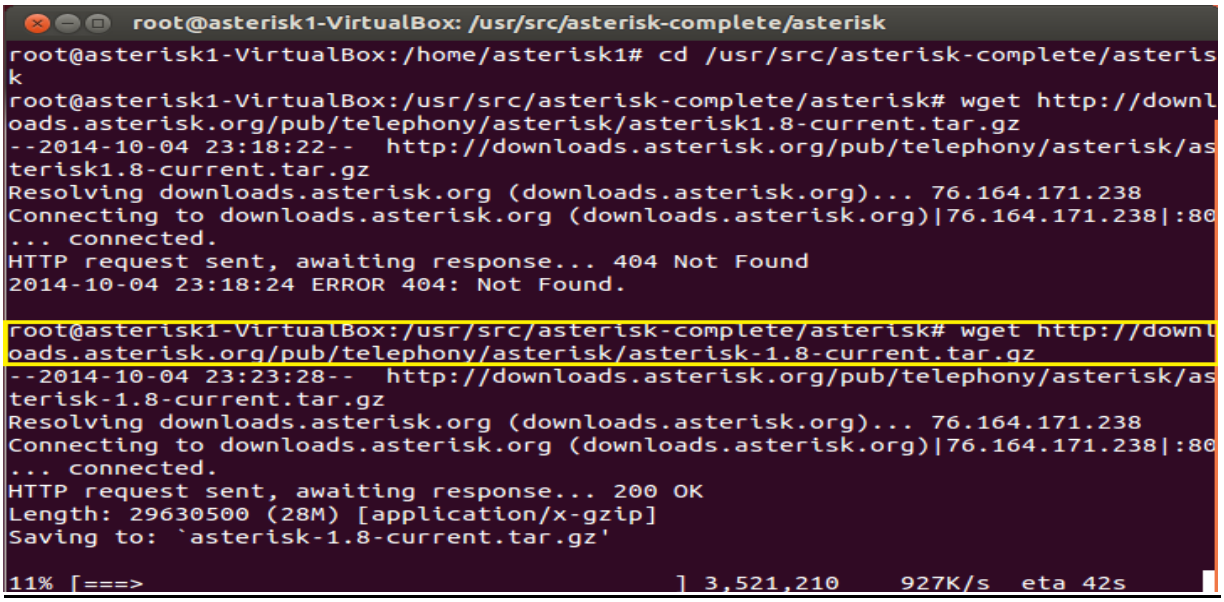


Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Se descarga el código fuente:

wget http://downloads.asterisk.org/pub/telephony/asterisk/asterisk1.8-current.tar.gz

Ilustración 48. Descargar código fuente central Asterisk.

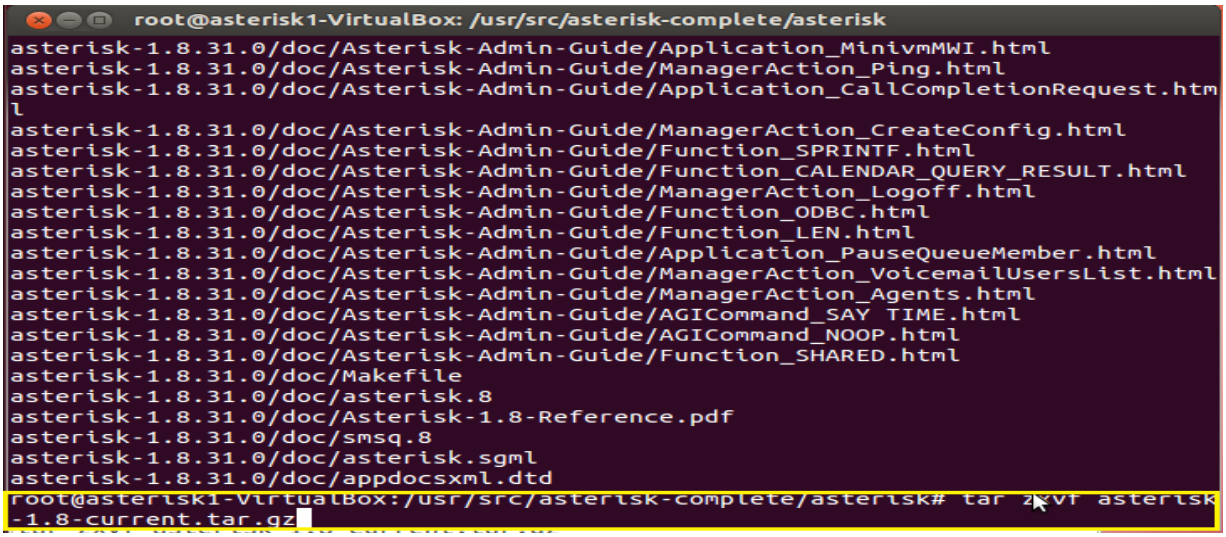


Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Se descomprime el archivo tar.gz descargado:

tar zxvf asterisk-1.8-current.tar.gz

Ilustración 49. Descomprimir el archivo Asterisk.

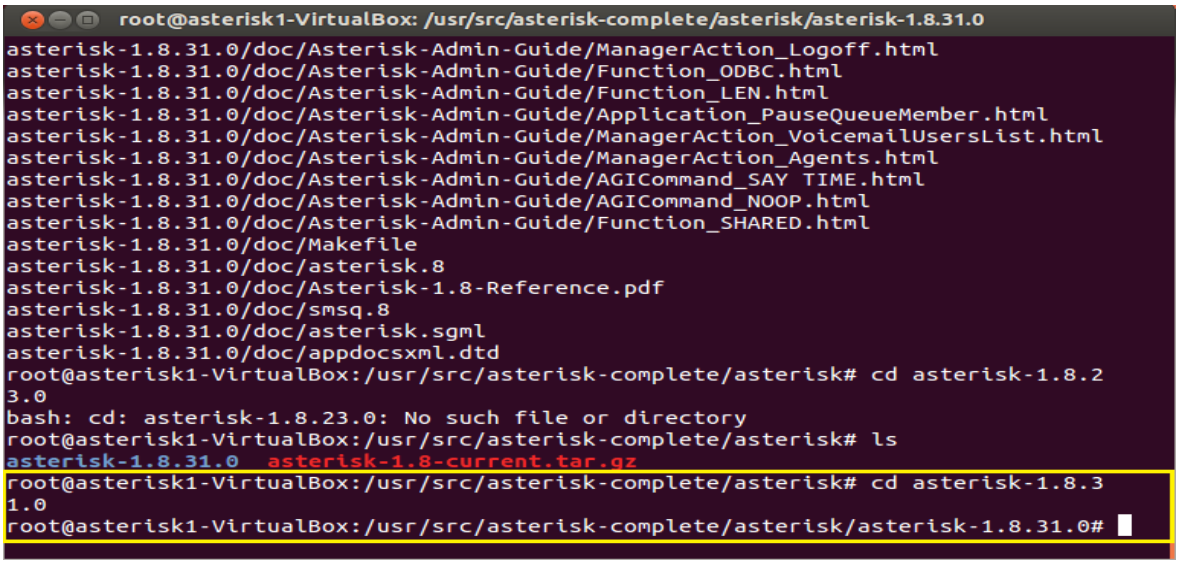


Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Se ingresa al directorio creado al descomprimir el archivo tar.gz:

cd asterisk-1.8.31.0

Ilustración 50. Ingresar carpeta descomprimida.

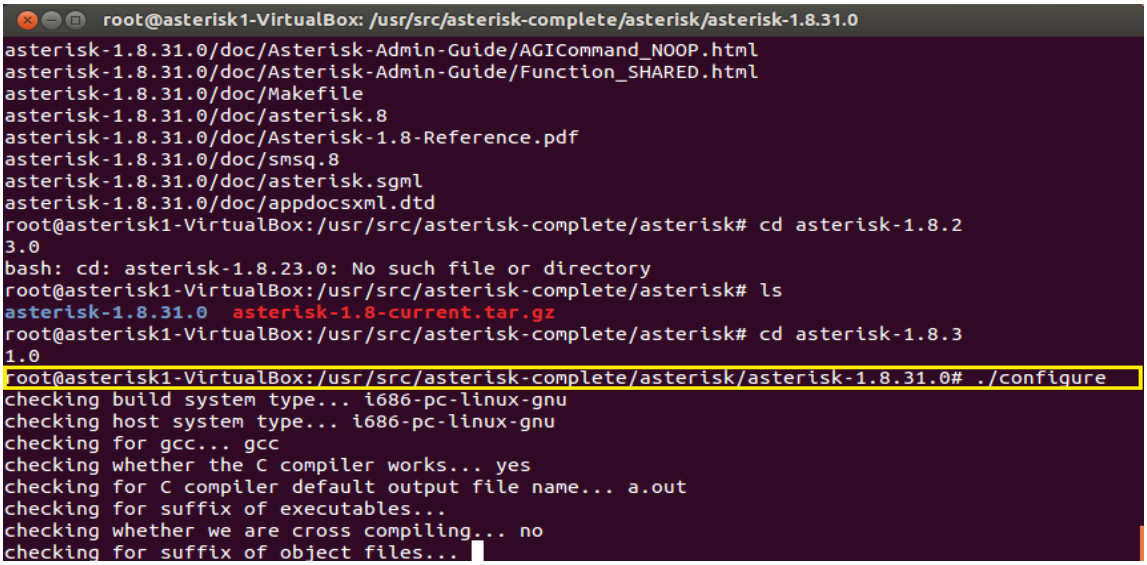


Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Y se ejecuta el script de configuración de Asterisk 1.8.31.0:

./configure

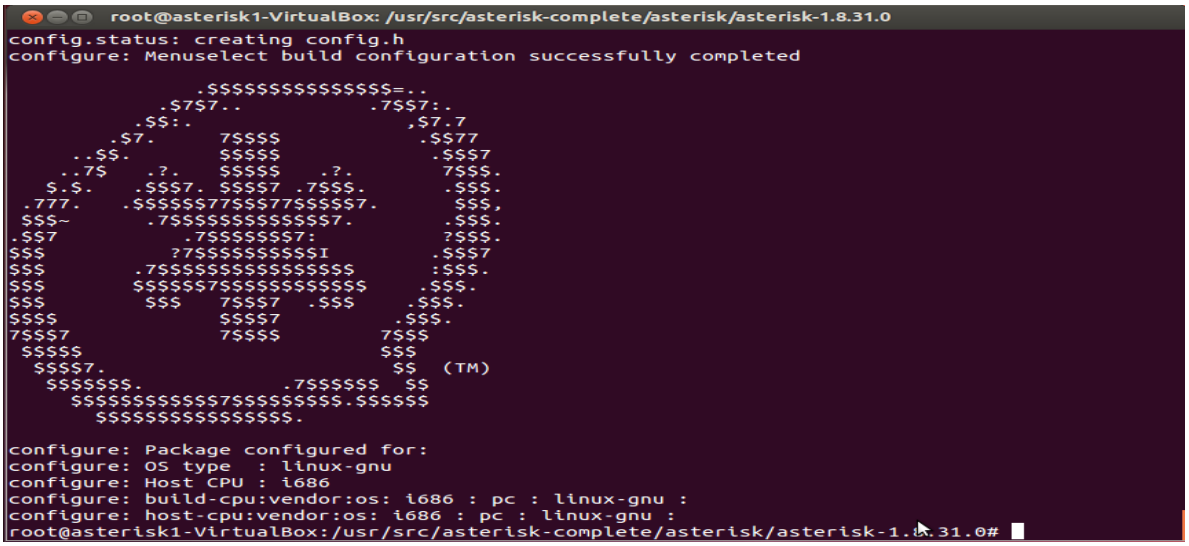
Ilustración 51. Ejecución script configuración Asterisk.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Cuando el proceso del script haya terminado deberá aparecer el símbolo que representa a la central Asterisk dentro de la terminal de comandos, como se observa en la figura:

Ilustración 52. Símbolo central Asterisk.

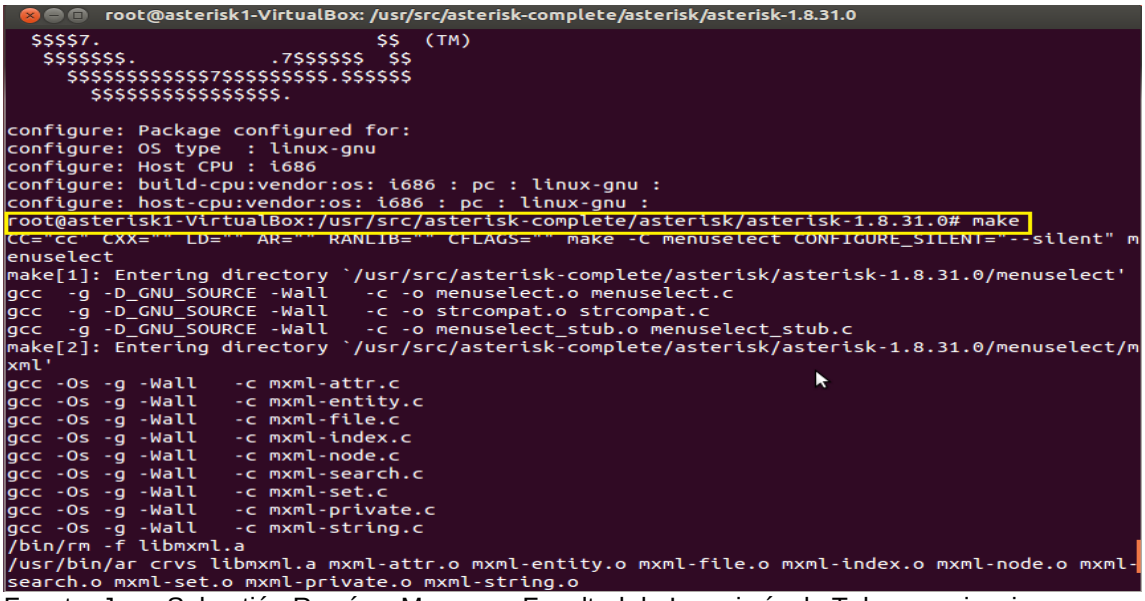


Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Ahora hay que compilar el código fuente para concluir con la instalación:

Make

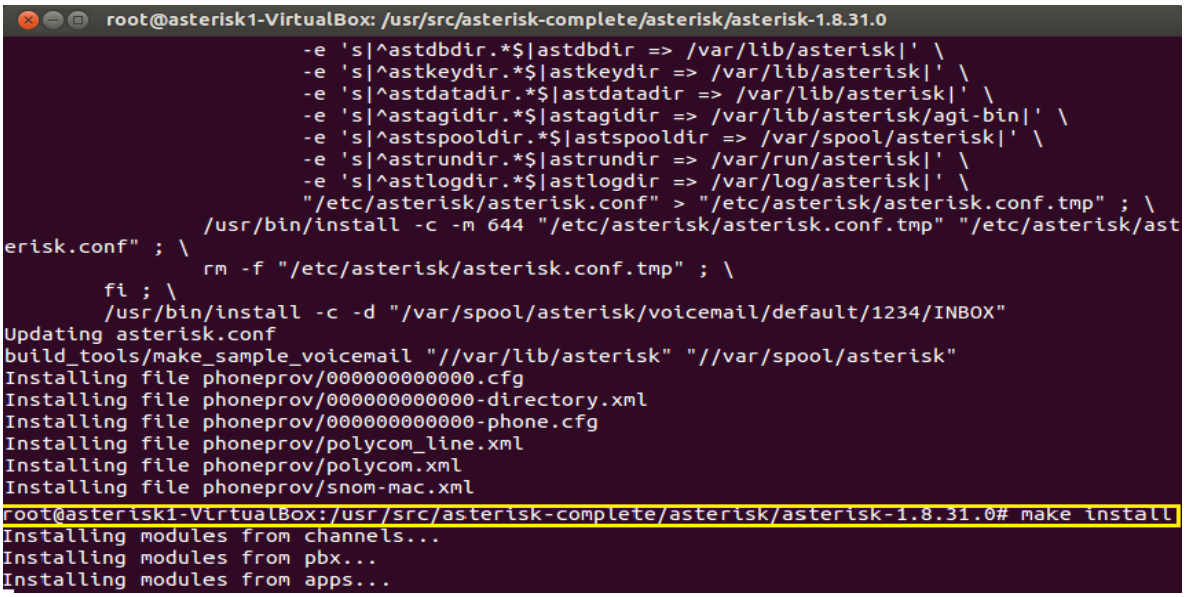
Ilustración 53. Compilación código fuente.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

make install

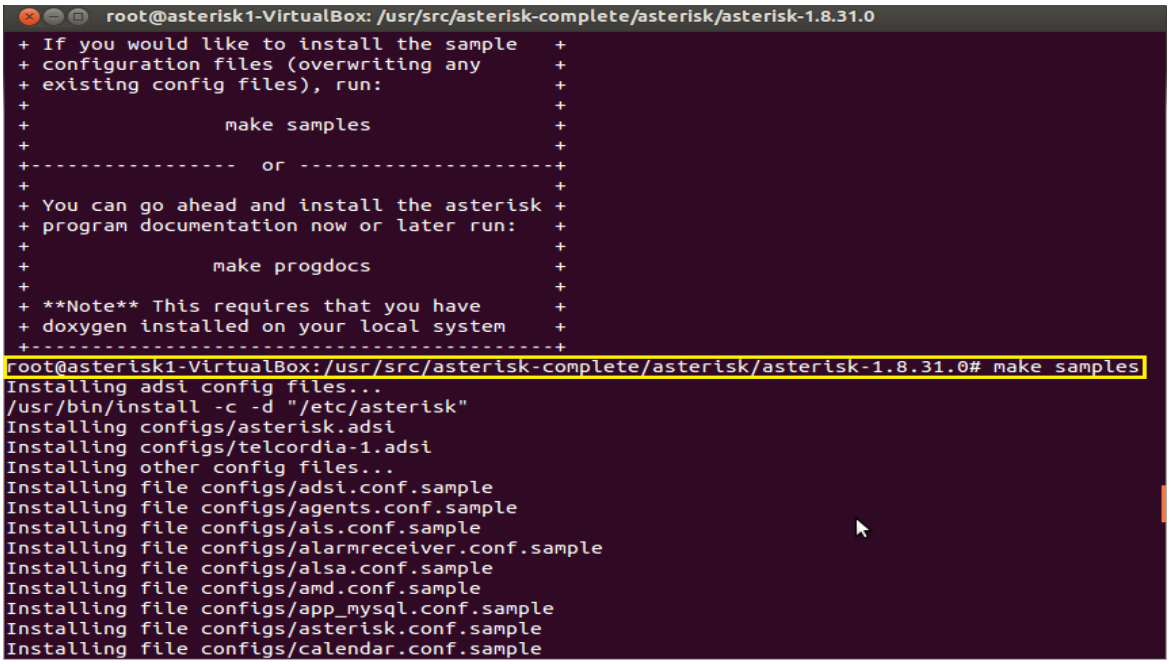
Ilustración 54. Compilación código fuente 2



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

make samples

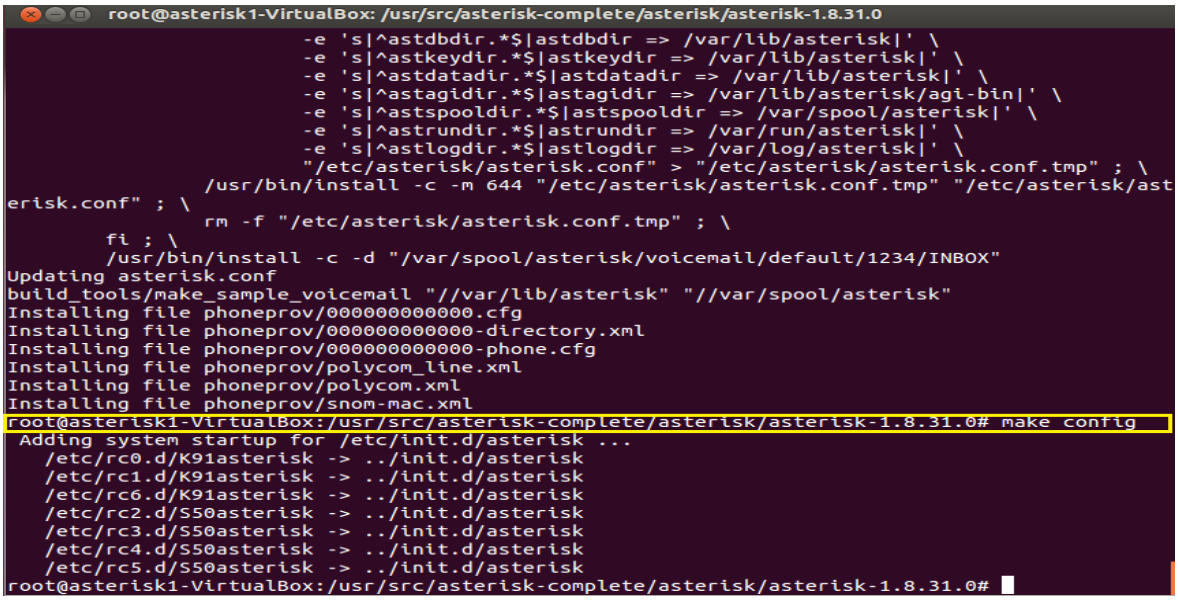
Ilustración 55. Compilación código fuente 3.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

make config

Ilustración 56. Compilación código fuente 4.

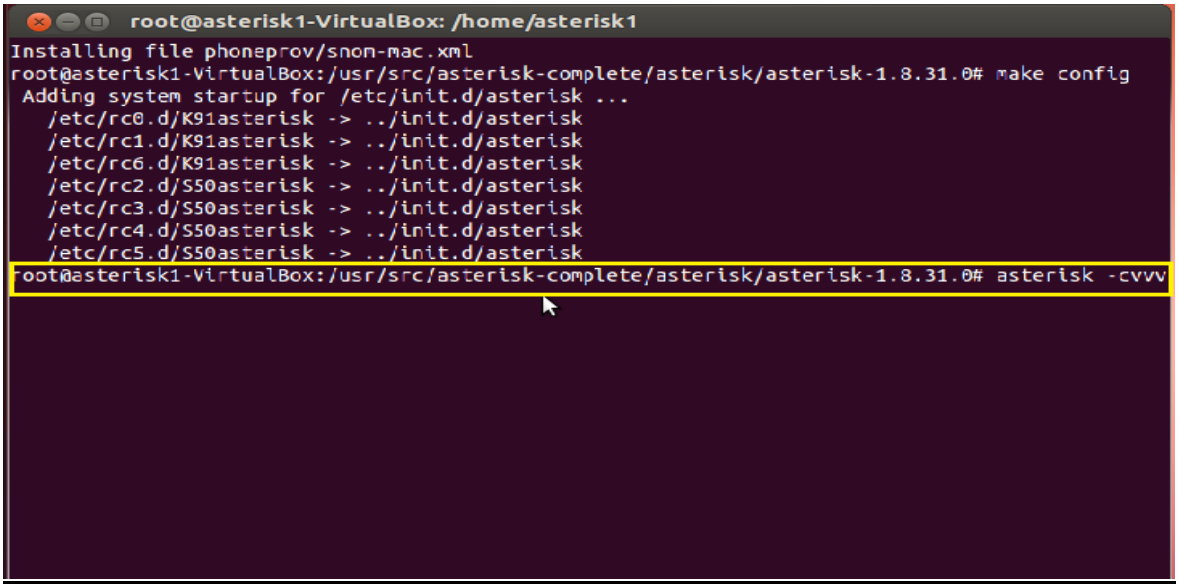


Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Después de compilar completamente el código fuente, la central Asterisk queda totalmente implementada y lista para configurar los servicios de voz IP. Para iniciar la central es necesario ejecutar el siguiente comando:

asterisk –cvvv

Ilustración 57. Inicio central Asterisk.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

- **Configuración de un servidor de nombres de dominio DNS para IPv6**

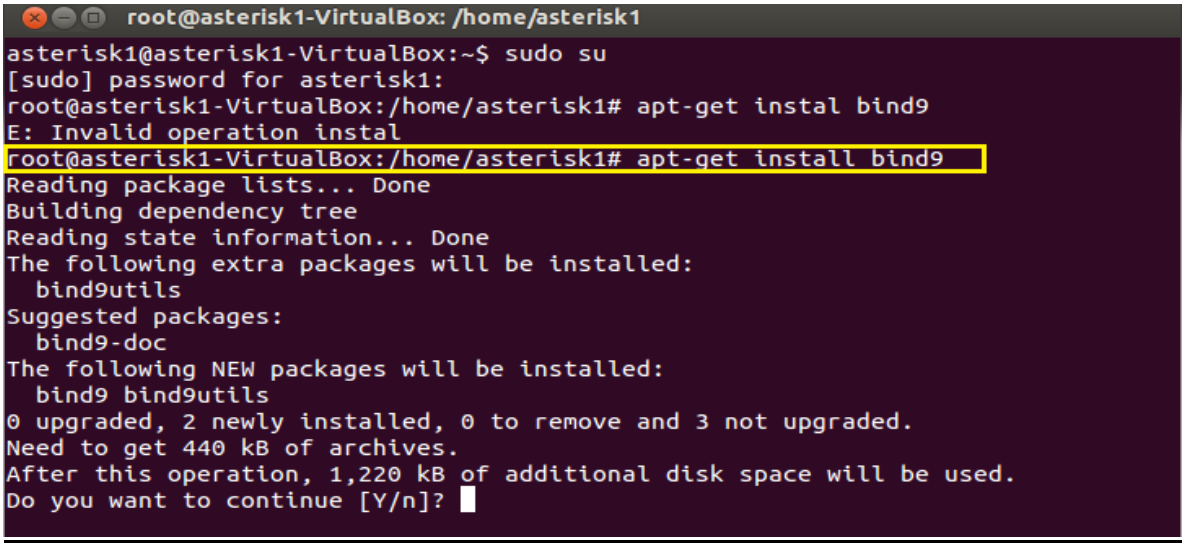
Para implementar un servicio de voz IP con el protocolo IPv6 es necesario utilizar un servidor de nombres de dominico conocido como servidor DNS (Domain Name Service). El servidor DNS se encarga de asignarle un nombre al servidor Asterisk en específico para evitar el problema de recordar las direcciones IP del servidor, en este caso el nombre que se le asigna a la central Asterisk es Asterisk1.rumbo.edu.co en vez de que los clientes estén recordando una dirección en IPv6. Gracias a esta ventaja este tipo de servidores son muy utilizados en las redes de internet.

- **Instalación de dependencias**

El servidor Asterisk esta implementado en el sistema operativo Ubuntu como se observa anteriormente, por tal motivo de utiliza la herramienta BIND (Barkeley Internet Naming Daemon)" siendo compatible con este sistema operativo. Esta herramienta es utilizada para implementar en servidor de nombres de dominio.
Para descargar la herramienta Bind9 hay que ejecutar el siguiente comando:

apt-get install bind9

Ilustración 58. Instalación herramienta BIND.



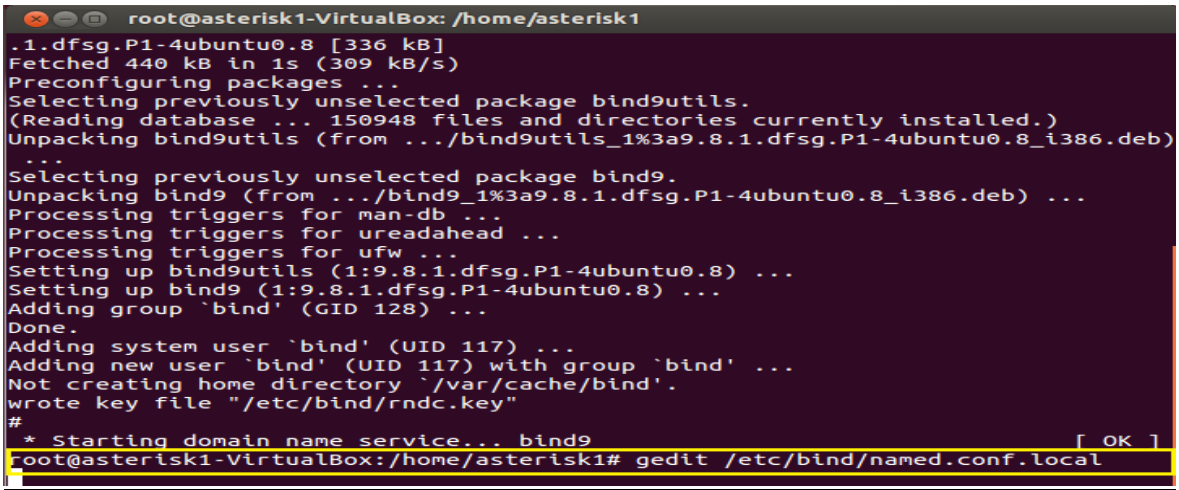
Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

- **Configuración del fichero de configuración de nombres DNS**

Para configurar el servidor de nombres de dominio hay que ingresar primero al fichero “named.conf.local”:

gedit /etc/bind/named.conf.local

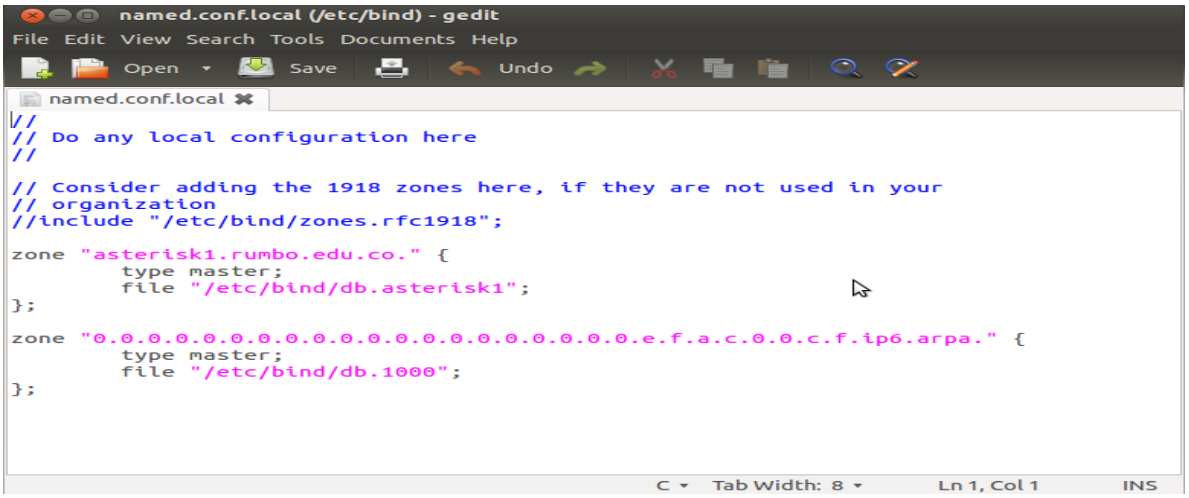
Ilustración 59. Ingreso al fichero Bind/named.conf.local.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Estando dentro de editor para la configuración de nombre de dominio se debe ingresar los siguientes parámetros, se determina si es un servidor primario o maestro, se agrega la zona de dominio del servidor DNS, a continuación se especifica la ubicación del fichero donde esta las configuración del servidor de nombres, por último se configura la resolución inversa del nombre de dominio, esto lo utiliza el servidor para resolver el nombre de dominio que le corresponde a una dirección IP. En este caso, como se está trabajando con direccionamiento IPv6 la zona inversa de dominio pasa de ser ".in-addr.arpa" en IPv4 a ser ".ip6.arpa." para direcciones IPv6.

Ilustración 60. Parámetros de configuración nombre de dominios.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Como se observa en la imagen los segmentos de la dirección inversa IPv6 están separados cada cifra hexadecimal por puntos y al finalizar la dirección la configuración de la resolución inversa para direcciones IPv6.

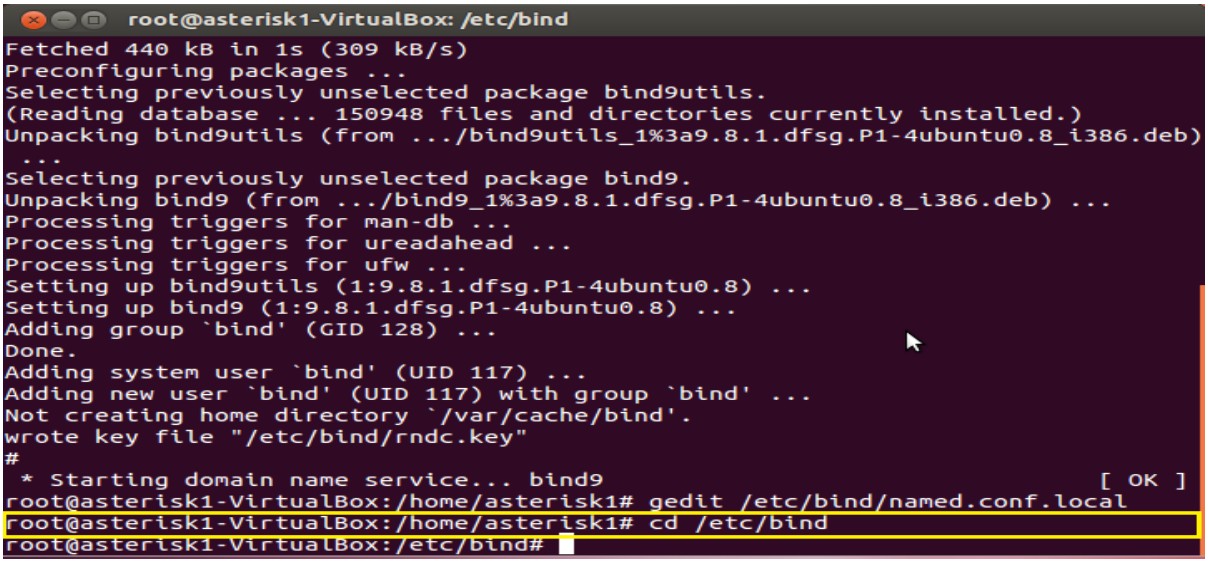
Se guardan los cambios hechos al fichero y luego se sale del mismo.

- **Configuración de ficheros para las zonas del servicio DNS**

Primero es necesario entrar al directorio de Bind9 para crear los archivos de configuración:

cd /etc/bind

Ilustración 61. Ingreso directorio Bind9.

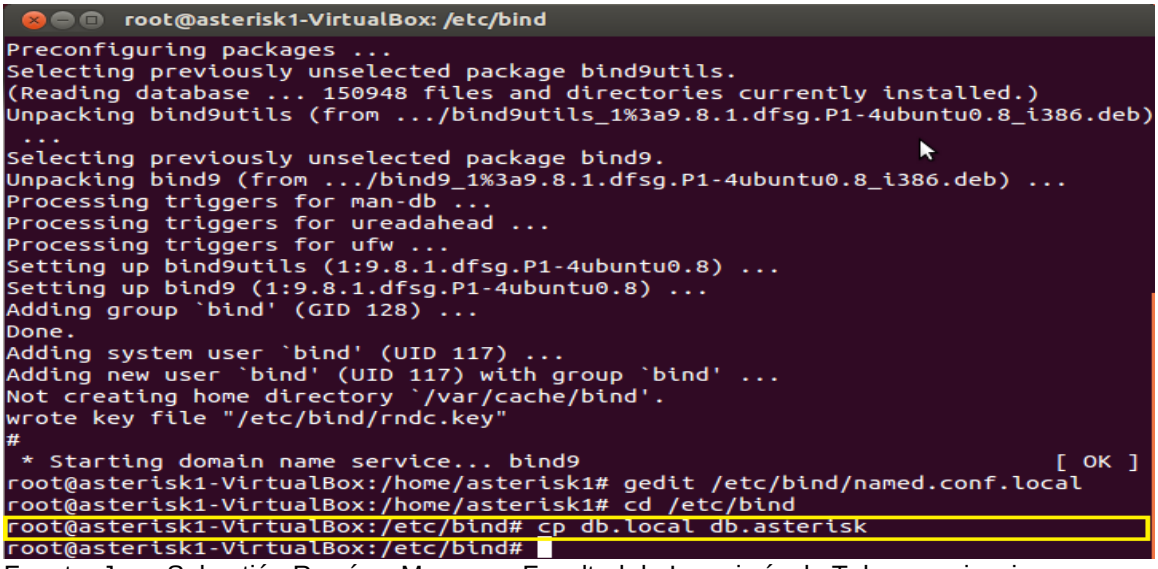


Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Al momento de la instalación de la herramienta Bind9 también se instalan por defecto unos archivos de los cuales se copian dos “db.local” y “db.127” a los ficheros establecidos dentro de la configuración de las zonas del servidor de nombres:

cp db.local db.asterisk

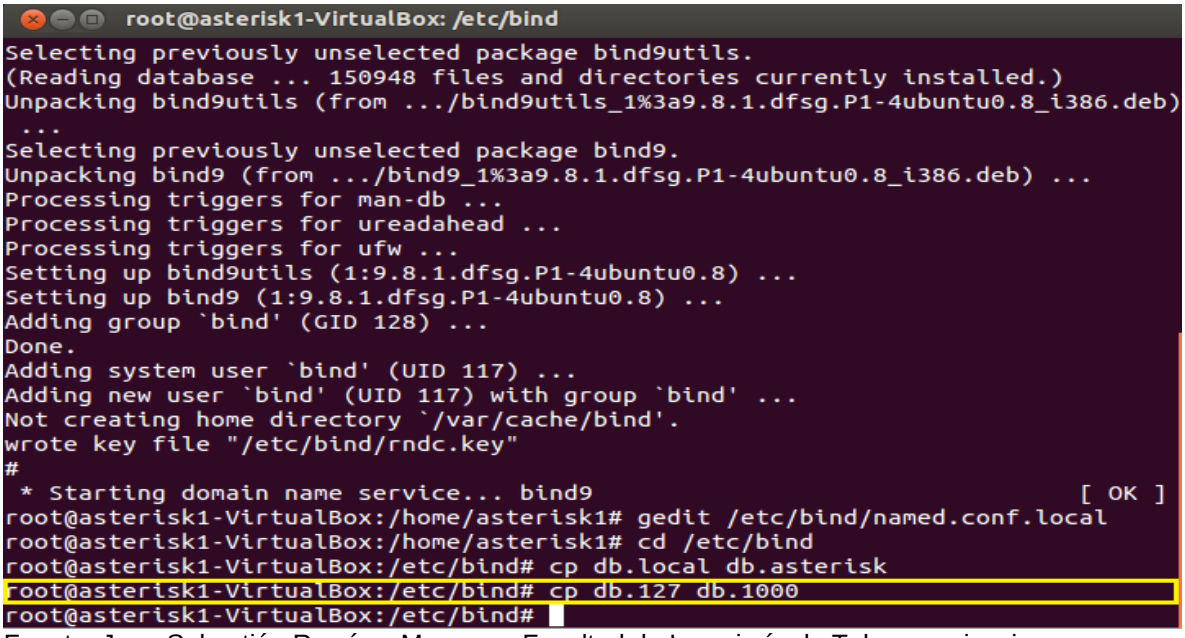
Ilustración 62. Copia archivo db.local db.asterisk.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

cp db.127 db.1000

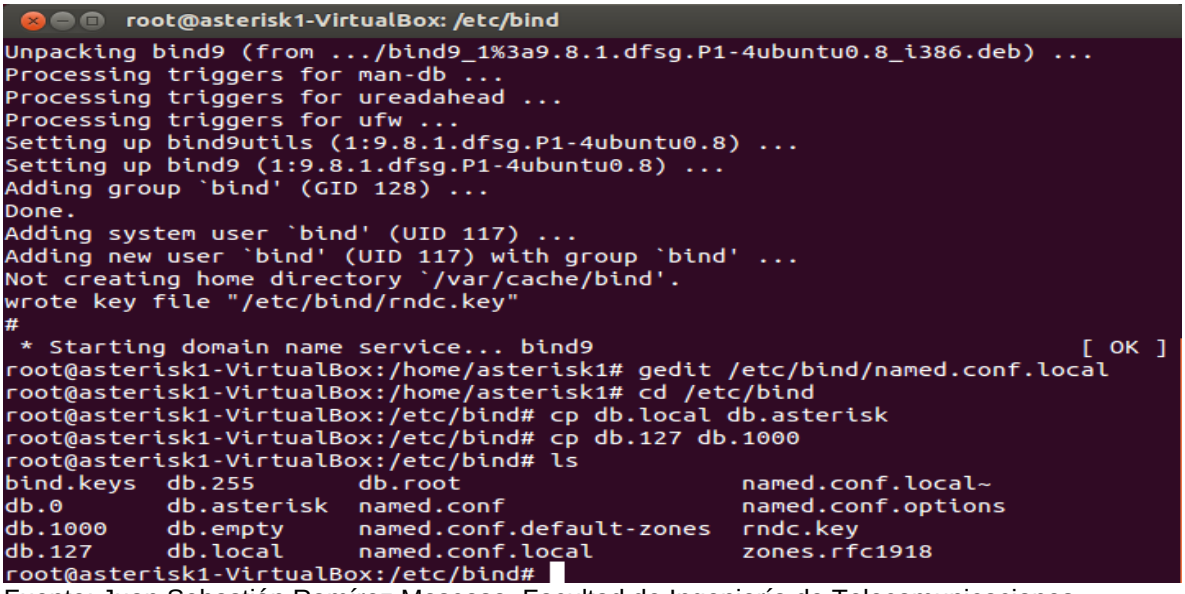
Ilustración 63. Copia archivo db.127 db.1000.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Como se observa en la siguiente imagen los archivos quedaron copiados dentro del fichero.

Ilustración 64. Archivos copiados en el fichero.



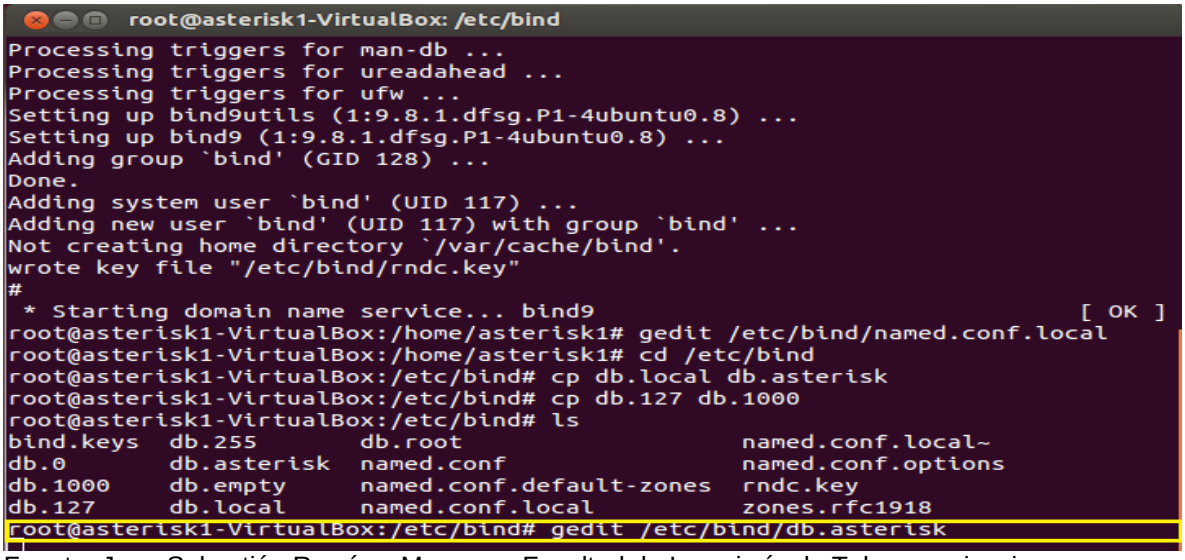
Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Se ingresa al archivo donde se darán las configuraciones a la zona directa del

DNS:

gedit /etc/bind/db.asterisk

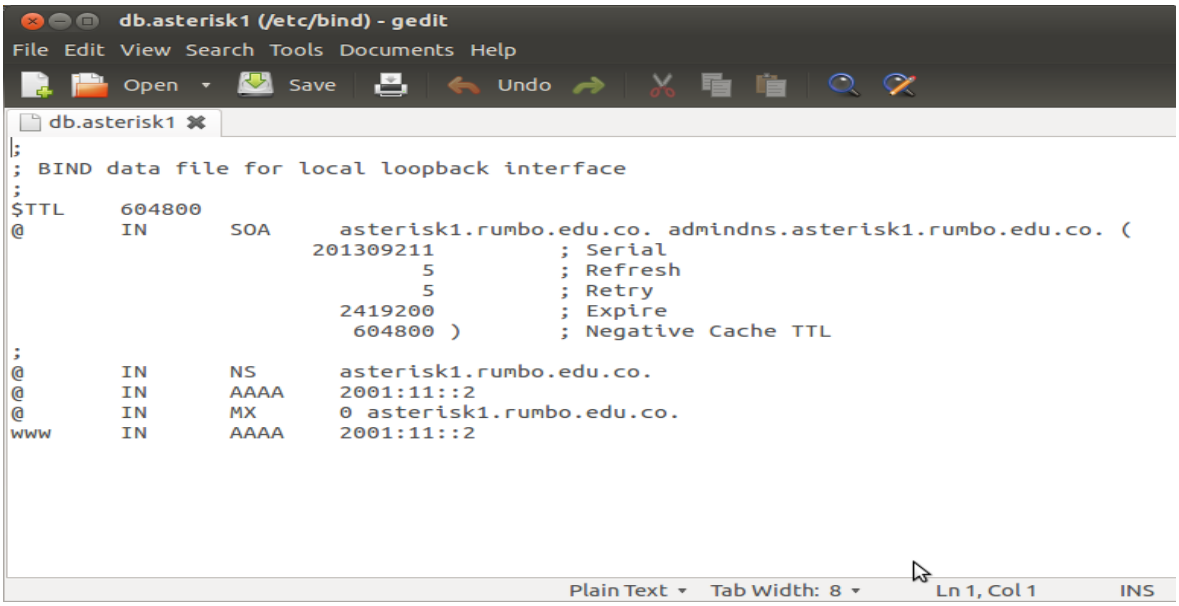
Ilustración 65. Ingreso archivo configuración zona directa db.asterisk.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

En el editor para la configuración de la zona directa para el servidor de nombres se colocan los siguientes parámetros:

Ilustración 66. Parámetros de configuración nombre de dominios.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

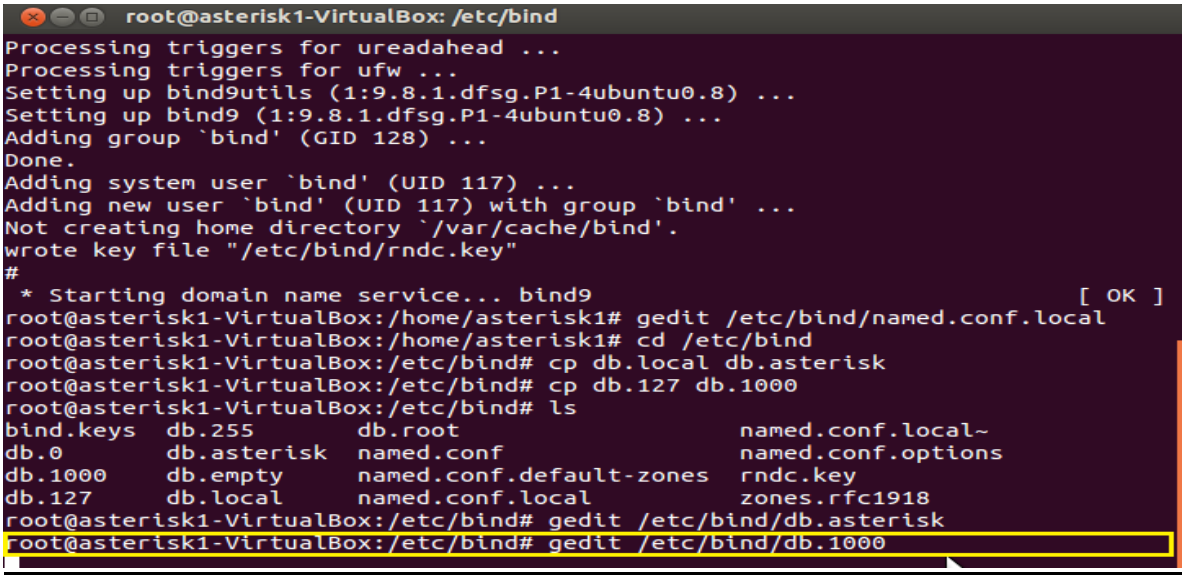
Como se observa en la imagen anterior se define el nombre de dominio que en este caso es asterisk1.rumbo.edu.co., también se observa el número serial el cual debe coincidir con el serial del archivo de la zona de resolución inversa, la configuración de la clase internet IN, los descriptores de registro, NS para el servicio de nombres, AAAA para la descripción de una dirección IPv6, MX que establece el intercambio de mensajes y el alias www que permite ingresar al servidor de nombres con el alias de www.asterisk1.rumbo.edu.co.

Al terminar la configuración se guarda y se cierra el editor.

Se ingresa al fichero de zona inversa:

gedit /etc/bind/db.1000

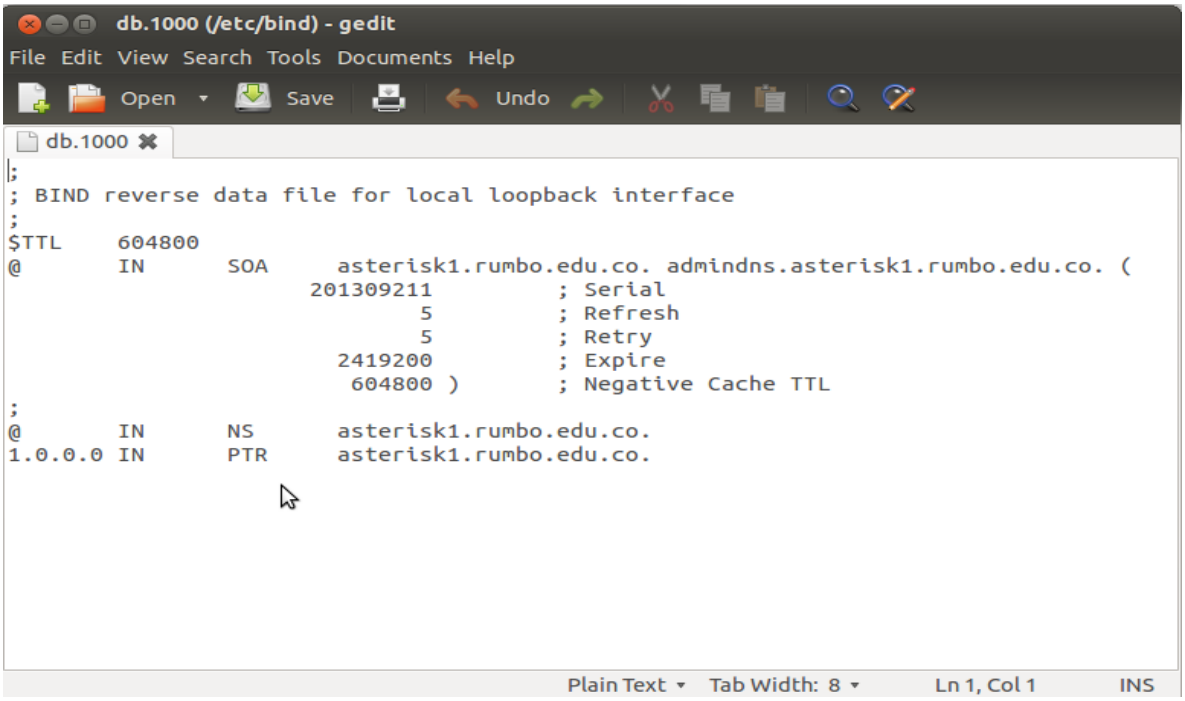
Ilustración 67. Ingreso archive Configuración zona directa db.1000



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Al ingresar al editor se debe colocar el número de serial igual al que se estableció en el archivo db.asterisk1, también se estable el nombre de dominio y se ingresa el registro NS para el registro inverso de nombre de dominio:

Ilustración 68. Parámetros de Configuración zona directa db.1000.

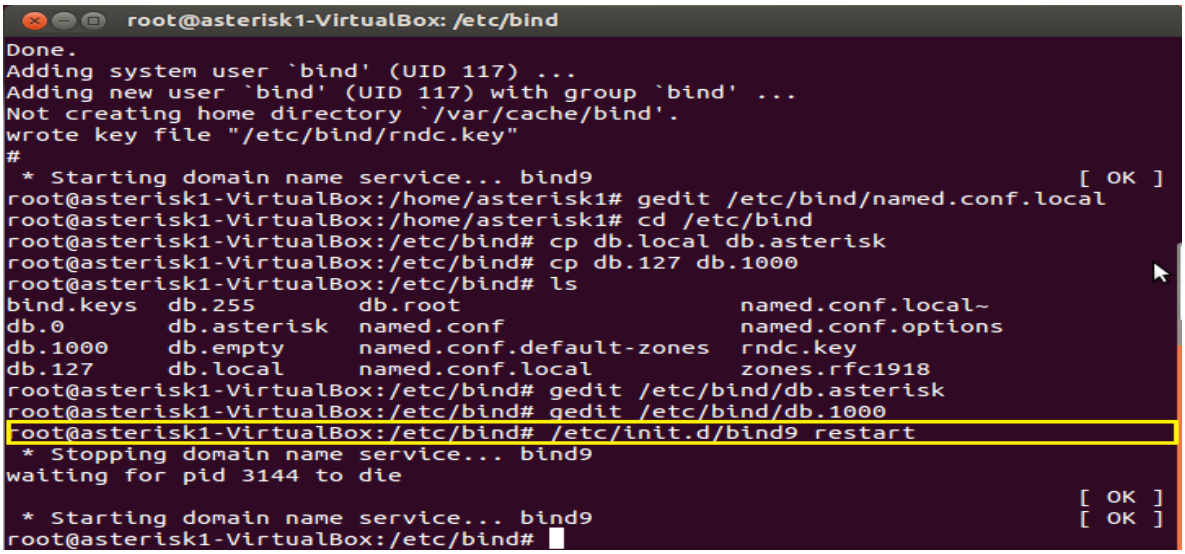


Parámetros de Configuración zona directa db.1000. Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Se guarda la configuración realizada y se reinicia el servicio Bind9:

/etc/init.d/bind9 restart

Ilustración 69. Reinicio servicio bind9.



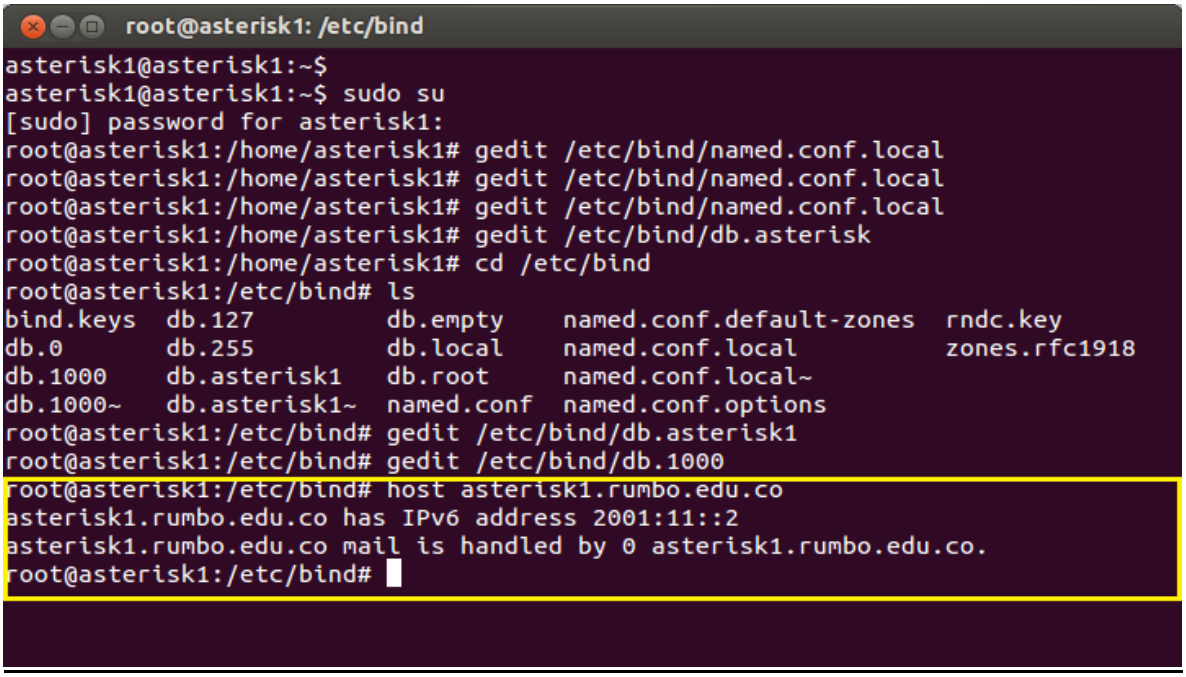
Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

- **Pruebas de funcionamiento del servicio de nombres de dominio**

Para comprobar que el DNS quedo configurado de manera correcta y que está funcionado sin inconvenientes se utiliza el comando host para observa la resolución del servidor de forma directa o inversa, como resultado se puede observar las líneas encerradas en el recuadro amarillo la resolución de forma directa, del mismo modo se realiza la operación para la forma inversa.

host asterisk.rumbo.edu.co

Ilustración 70. Prueba servidor DNS



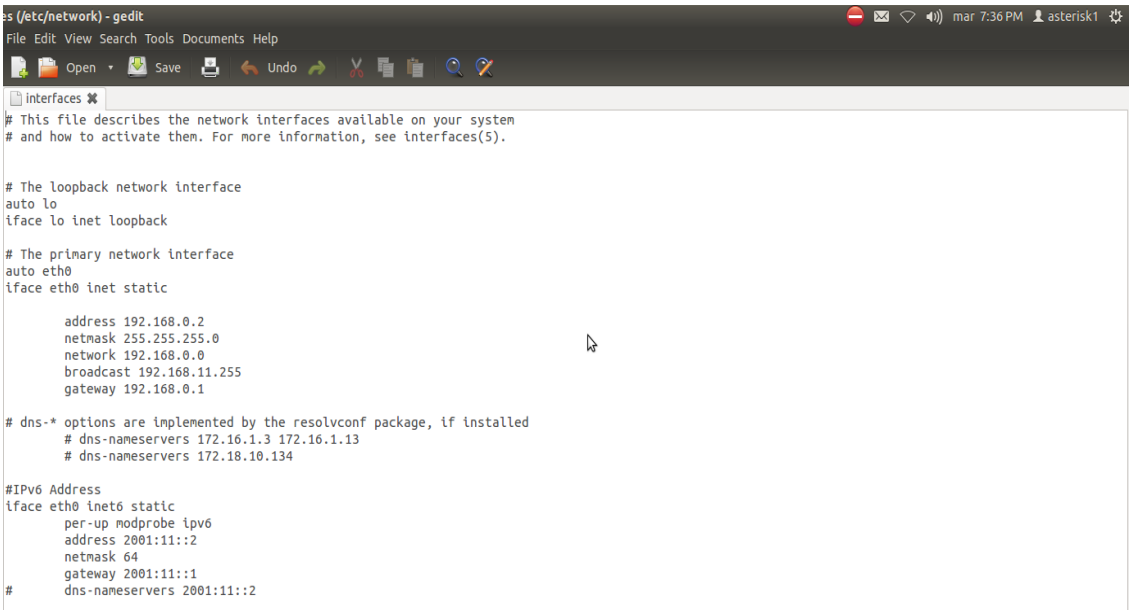
Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

- **Configuración de red del servidor Asterisk**

Se ingresa al fichero de configuración de la interfaz de red del equipo:

gedit /etc/network/interfaces

Ilustración 71. Configuración interfaz central Asterisk.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Como se puede observar en la anterior imagen se configuran los parámetros para la interfaz del servidor. Se ingresa la dirección IPv4 e IPv6, la máscara de red para las dos versiones del protocolo IP, la red a al que pertenece el equipo, la dirección de difusión, la compuerta de la red tanto para IPv4 como para IPv6 y el servidor de nombre de dominios para el caso de IPv6. Se aclara que para este caso se está configurando la interfaz eth0, asignándole direccionamiento estático.

Se guarda la configuración realizada, se sale del editor y por último se reinicia las interfaces del servidor con el siguiente comando:

/etc/init.d/networking restart

4.2.1.2 CONFIGURACIÓN DE RED DE WINDOWS

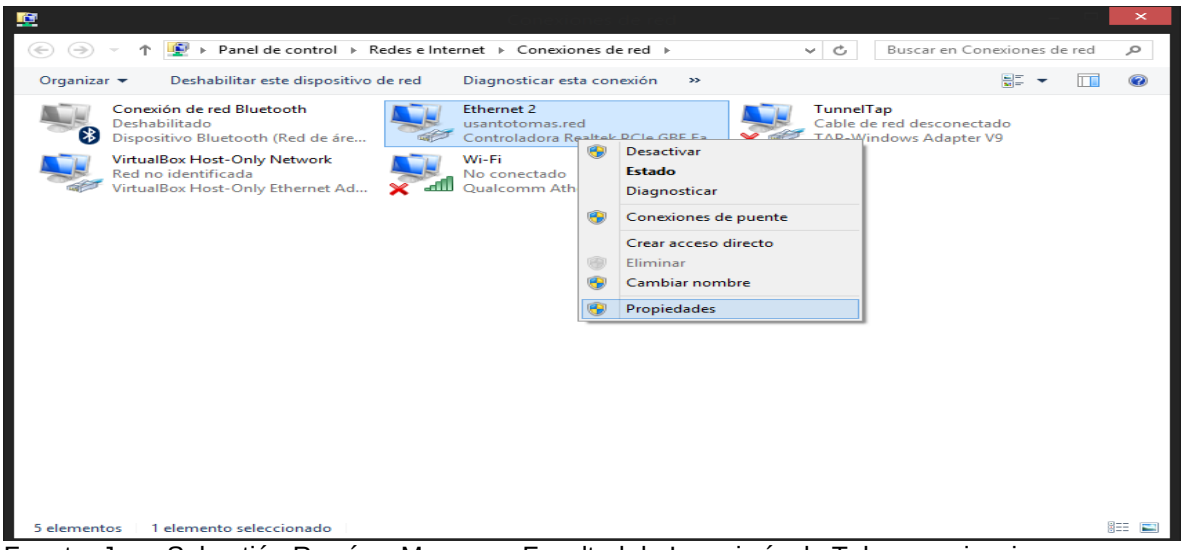
Para el desarrollo del proyecto se utilizó un equipo con sistemas operativo Windows 7, instalándole el softphone para realizar llamadas entre el teléfono IP y este. Se utiliza este sistema operativo ya que funciona tanto con IPv4 como IPv6 siendo una gran ventaja para la realización de los experimentos.

Para configurar la interfaz se ingresa por la siguiente ruta:

Panel de control\Redes e Internet\Conexiones de red

Se elige la interfaz física del equipo, se presiona clic derecho sobre la interfaz y se selecciona la opción propiedades para configurar esta:

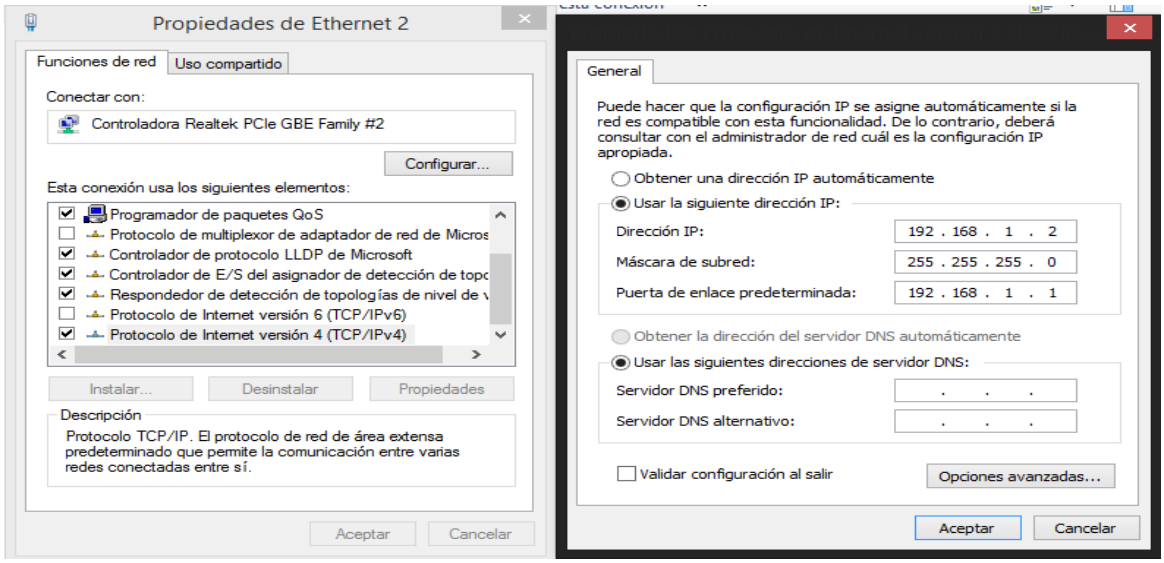
Ilustración 72. Configuración interfaz Windows.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

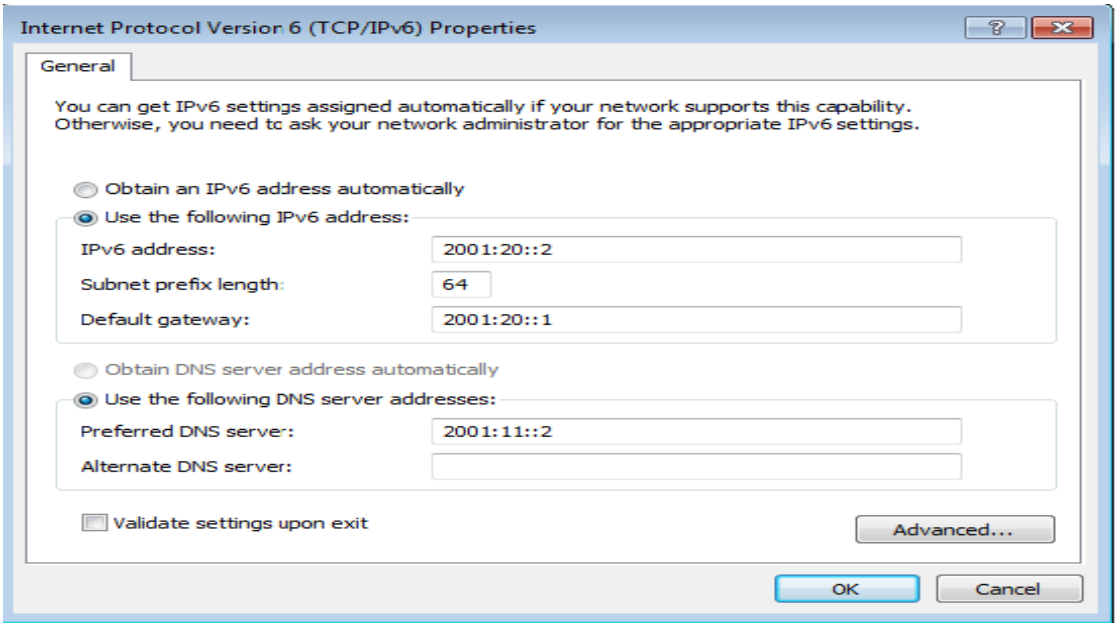
Se configuran los dos protocolos tanto IPv4 como IPv6, estableciendo la dirección IP, la máscara, la puerta de enlace y la dirección del servidor de nombres de dominio para el caso de IPv6. Los protocolos IPv4 e IPv6 pueden convivir sin ningún problema en la interfaz al mismo tiempo.

Ilustración 73. Configuración IPv4.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Ilustración 74. Configuración IPv6



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

4.2.1.3 CONFIGURACIÓN DE TELÉFONOS IP CORPORATIVOS GRANDSTREAM GXP 1400

GXP 1400 son teléfonos corporativos utilizados en pequeñas y medianas empresas brindando muy buenas funcionalidades. Entre las características con las que cuenta se puede encontrar protección de seguridad avanzada, cuenta con dos líneas con cuentas SIP cada una, excelente calidad de audio, interoperabilidad con otros dispositivos, redes y plataformas, y soporta direccionamiento para los dos protocolos IP (IPv4 e IPv6).

El teléfono GXP cuenta con las siguientes especificaciones técnicas:

- Cuenta con los siguientes protocolos: SIP RFC3261, TCP, IP, UDP, RTP, HTTP/HTTPS, ARP/RARP, ICMP, DNS, DHCP, PPPoE, TELNET, TFTP, NTP, STUN, SIMPLE, TR-069, 802.1x.

- Tiene dos puertos de conmutación 10/100Mbps.

- Soporta los codecs de voz G.723.1, G.729A/B, G.711µ/a, G.726-32, G.722.

- En entorno de seguridad cuenta con contraseñas de usuario y de administrador, MD5 y MD5 basada en autenticación sess, encripción AES, SRTP, TLS y control de acceso al medio 802.1x.

- Potencia y eficiencia de energía con un adaptador de energía con entrada de 100-240VAC y frecuencia de 5060Hz, Salida de 5V DC, 800mA, consumo máximo de energía de 2,5W.

Estado de configuración de teléfono GXP1400

El teléfono IP GXP 1400 cuenta con un entorno grafico vía web por medio de la dirección IP que tenga establecida, para realizar la configuración de manera más fácil, para poder ingresar al entorno grafico se debe estar en la misma red que el teléfono. En el pantallazo inicia se solicita contraseña que en este caso en rumbo 2340.

Ilustración 75. Pantalla de autenticación para configuración del teléfono**.**

Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Al ingresa al modo grafico del teléfono podemos encontrar la información de red y la información del sistemas con la que cuenta el teléfono, esta información la encontramos en las pestañas ubicadas a mano izquierda con los botones Estado Información de la red y Estado Información del Sistema.

Configuración de red IPv4

Para configurar la dirección IPv4 se debe ir a la petalla que se encuentra en la parte de arriba de la pantalla que dice red y después la opción configuración básica. Se ingresa la dirección IP, la máscara, puerta de enlace y la dirección IP del servidor Asterisk. Se debe elegir en la parte superior la opción Preferir IPv4, por último se guarda y se aplican los cambios.

Ilustración 76. Configuración IPv4 para el teléfono.

Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Configuración de red IPv6

Para configurar la dirección IPv6 se cambia el protocolo a utilizar, se selecciona las opciones Preferir IPv6 y Completamente Estática, se asigna la dirección IP y el tamaño del prefijo, por último se guarda y se aplican los cambios.

Ilustración 77. Configuración IPv6 para el teléfono.

Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Se aclara que el teléfono funciona dual stack con los dos protocolos, pero al momento de registrar la cuenta SIP se realiza con el protocolo que se ha seleccionado como preferido en la configuración anterior.

Configuración de una cuenta SIP en IPv4

Para configurar la cuenta SIP en IPv4 se debe elegir en la pestaña cuentas, eligiendo algunas de las dos cuentas que están disponible, en este caso se selecciona la cuenta 1 y después Configuraciones generales, luego se activa la cuenta en Cuenta Activa, se ingresa la dirección IP del servidor SIP y del proxy de salida, se ingresa el identificador del usuario y se autentica, por último se ingresa la contraseña del usuario rumbo2340 previamente configurada en el archivo SIP. Se guardan y aplican los cambios.

Ilustración 78. Configuración de la cuenta SIP.

Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Configuración de una cuenta SIP en IPv6

En este caso para configurar la cuenta SIP para IPV6 se utiliza la cuenta 2, se activa y automáticamente se desactivando la cuenta 1, se configuran los parámetros ingresando la URL del servidor SIP IPv6 la cual está definida por el servidor de nombres, identificador del usuario y autenticarlo, se ingresa la clave rumbo2340 por último se guardan y se aplican los cambios realizados.

Se aclara que no se utiliza la dirección IPV6 para que el teléfono se registre con la cuenta SIP ya que este no reconoce direcciones IPV6 por lo que se le dio solución de la forma más viable que fue implementando un servidor de nombres de dominio (DNS) para direccionamiento IPv6, para este caso el nombre de dominio utilizado es Asterisk1.rumbo.edu.co con el que se registra el teléfono para brindar servicios de voz IPv6 nativa.

Ilustración 79. Configuración de la cuenta SIP IPv6.

Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Estado de las cuentas SIP

Cuando se realiza la configuración de manera correcta tanto para IPv4 como para IPv6 se mostrara en la configuración del teléfono un SI resaltado de color verde indicando que el registro fue exitoso, dependiendo la cuenta que se haya prefiera, también el modo grafico del teléfono indica con una bocina que el registro del teléfono en el servidor Asterisk se realizó y está disponible para realizar llamadas.

Ilustración 80. Estado de la cuenta SIP IPv4 activa.

Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Como se mencionó anteriormente para el caso de la cuenta dos el registro se realiza por el servidor DNS, se muestra que mismo SI resaltado de color verde para mostrar que la cuenta 2 se registró de manera correcta.

Ilustración 81. Estado de la cuenta SIP IPv6 activa.

Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

4.2.1.4 CONFIGURACIÓN DE SOFTPHONES

Un softphone es un software que emula un teléfono IP virtual instalado en un computador, que permite realizar llamadas entre softphones y de softphone a teléfonos IP. Se registra ante el servidor Asterisk como un teléfono IP convencional, asignándole una extensión y prestando servicios en tiempo real como VoIP, video video llamas, video conferencias, etc.

Para el desarrollo de este proyecto se utilizaron dos tipos de softphone, el primero que se utilizó fue Zoiper 3.6 para el protocolo IPv4, el segundo que se utilizó fue Linphone 3.6.1 para el protocolo IPv6 ya que el primero no cuenta con la característica de registro para IPv6.

Zoiper 3.6

Zoiper es un software de licencia libre, con funcionalidades muy buenas como el soporte de los dos protocolos de señalización de Asterisk como lo son SIP e IAX, buena calidad de audio y video, no utiliza tantos recursos de ordenador. Este softphone es muy conocido corporativamente, utiliza el protocolo IPv4.

Para configurar el softphone Zoiper para la realización de los experimentos se realizaron los siguientes pasos:

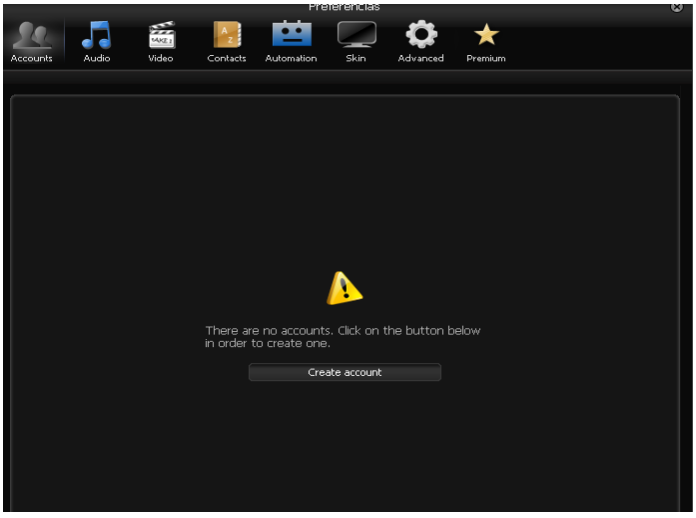
Primero se selecciona en la parte superior la pestaña que dice configuración y se selecciona la opción preferencias.

Ilustración 82. Ingreso configuración Zoiper.

Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Aparece una venta con un botón en el medio que tiene la opción de crear la cuenta, se pulsa para ingresar y configurar los parámetros de la cuenta.

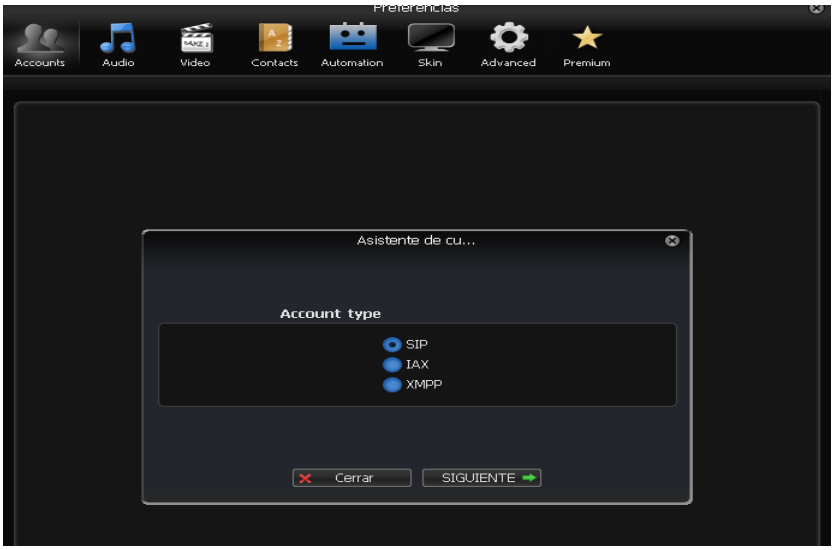
Ilustración 83. Creación cuenta SIP.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Después de ingresar se elige el tipo de cuenta que se desea utilizar, que para este caso se selecciona el protocolo SIP.

Ilustración 84. Elección cuenta SIP.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

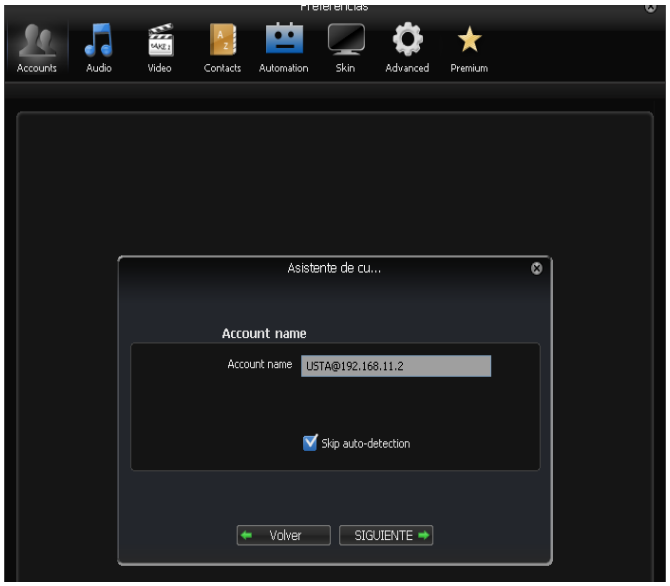
A continuación se configuran lo parámetros de la cuenta SIP para que el softphone realice el registro ante el servidor Asterisk. Se ingresan los datos del usuario o extensión, la clave de autentificación rumbo2340 y la dirección IP del servidor Asterisk.

Ilustración 85. Configuración cuenta SIP.

Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

La siguiente ventana que aparece después de configurar los parámetros muestra el nombre del servidor y también es para seleccionar si el softphone se registra automáticamente al servidor Asterisk, para este caso se elige esta opción.

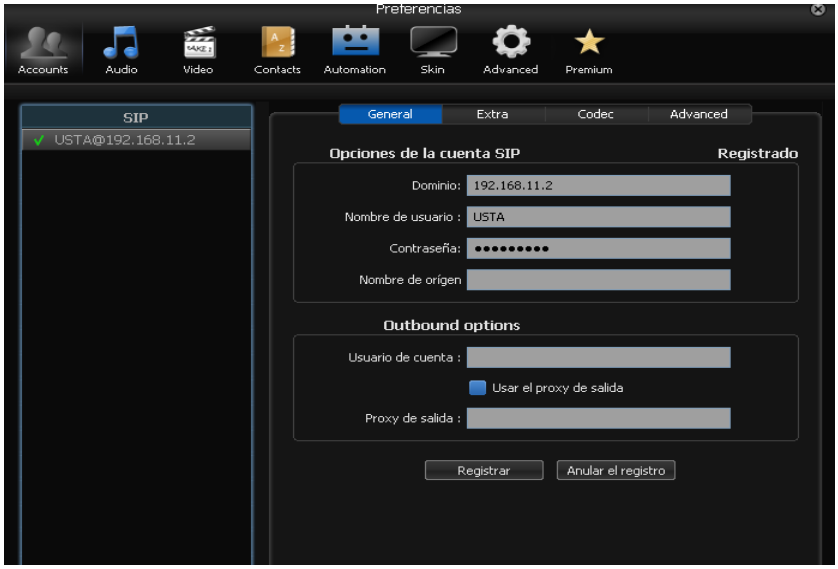
Ilustración 86. Nombre del servidor Asterisk.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Por ultimo muestra las configuraciones realizadas y parece un letrero en la parte superior derecho que dice registro indicando que el softphone se configuro de manera correcta y que está listo para realizar llamadas.

Ilustración 87. Registro softphone.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

En la figura siguiente se muestra el softphone funcionando con la extensión 103.

Ilustración 88. Zoiper funcionando.

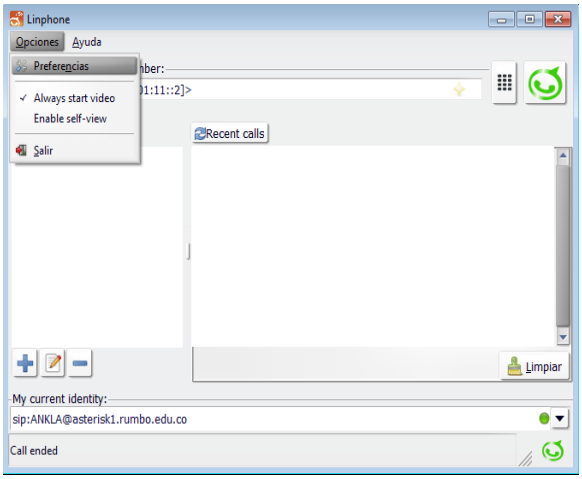
Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Linphone 3.6.1

Como el softphone anterior esta cuenta con licencia libre por lo que lo hace muy llamativo en los entornos empresariales por el bajo costo de implementación, es un software que soporta los protocolos de IP (IPv4 e IPv6). El registro de cuenta se produce mediante el protocolo de señalización SIP.

Para este proyecto se implementa gracias a su característica de brindar el servicio con el protocolo IPv6 nativa. Para realizar la configuración se selecciona la pestaña que se encuentra en la parte superior opciones, ingresando la opción preferencias.

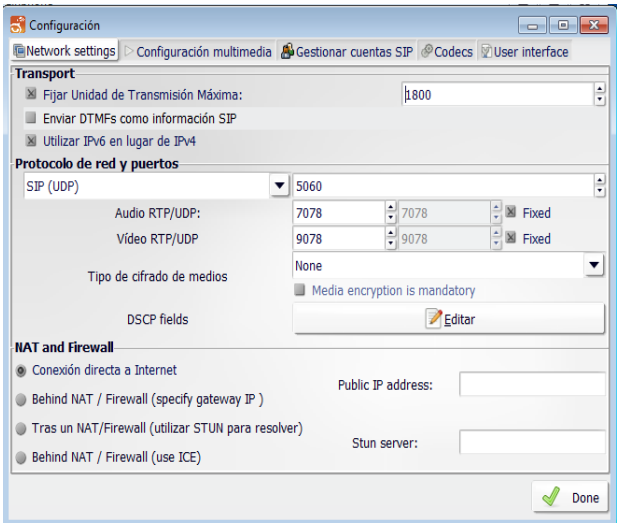
Ilustración 89. Ingresar a las configuraciones.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Se abre una venta de configuración donde se selecciona la opción que dice utilizar IPv6 en lugar de IPv4, habilitando el protocolo IPv6 para poder realizar el registro, a continuación se ingresa a la pestaña gestión cuenta SIP.

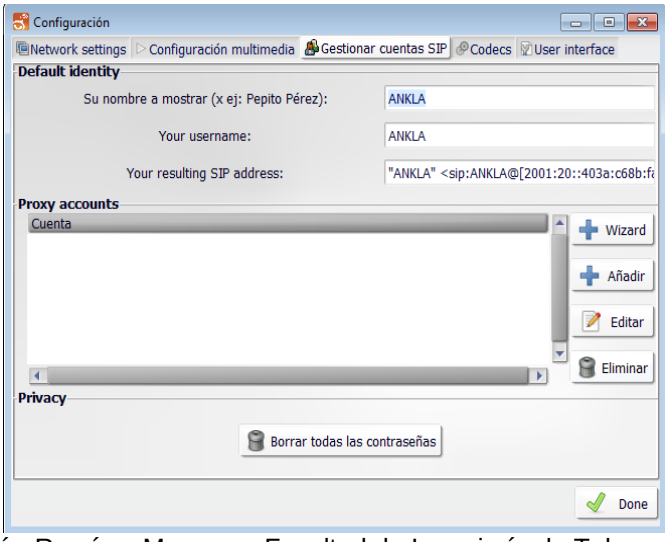
Ilustración 90. Opciones de configuración del softphone para IPv6.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Se ingresa el nombre de la extensión con la que el softphone va a funcionar registrando ante el servidor Asterisk. Se le Done para guardar la configuración y aplicarlas.

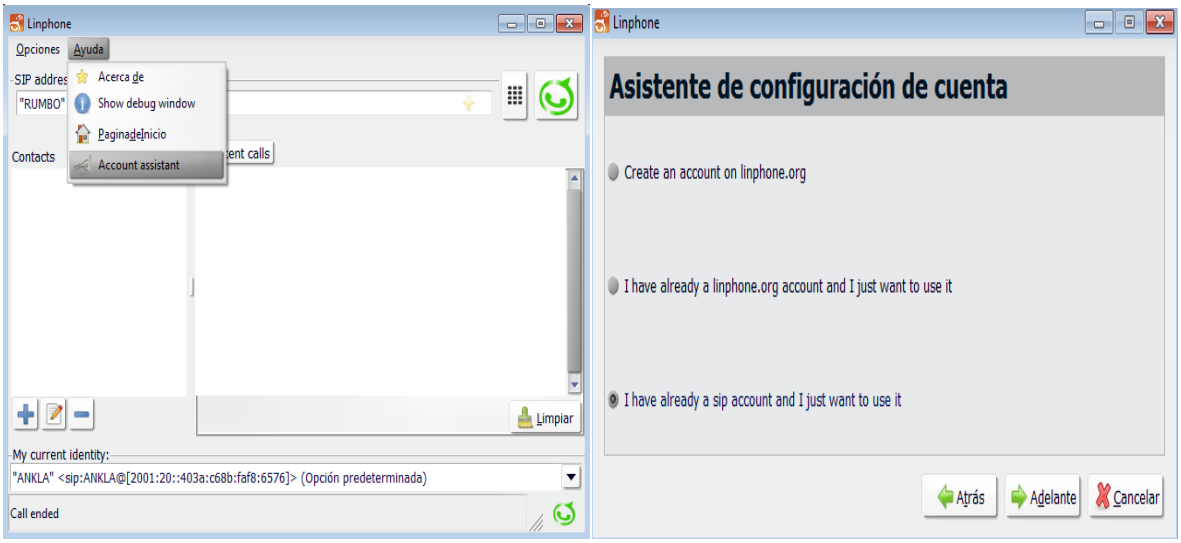
Ilustración 91. Gestión de cuentas SIP en el softphone



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Después de realizar la primera configuración, se debe ingresar a la pestaña localizada en la parte superior ayuda y se selecciona la opción asistente de cuenta, se abre una pestaña con tres opciones eligiendo la última para la creación de la cuenta SIP.

Ilustración 92. Asistente de configuración de cuentas SIP.

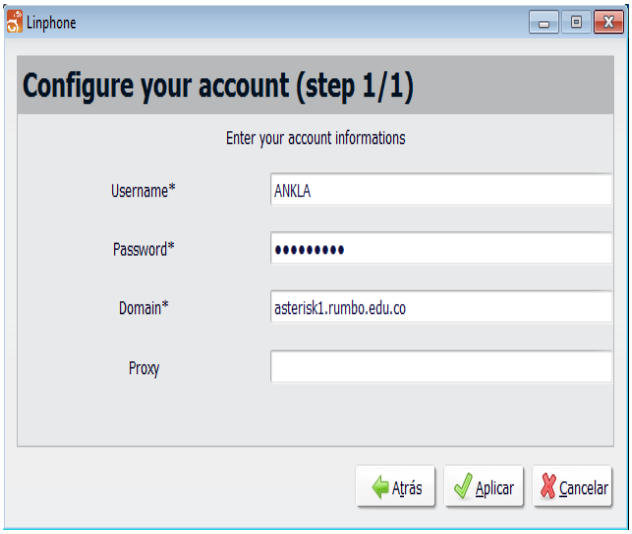


Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Se ingresan los parámetros para registrar la cuenta SIP ante el servidor Asterisk, se ingresa el usuario, la clave de autentificación rumbo2340 y para el caso de IPv6 no

se utiliza la dirección IP si no el domino que se le asignó al servidor DNS para realizar el registro en el servidor Asterisk, brindado el servicio de voz en IPv6.

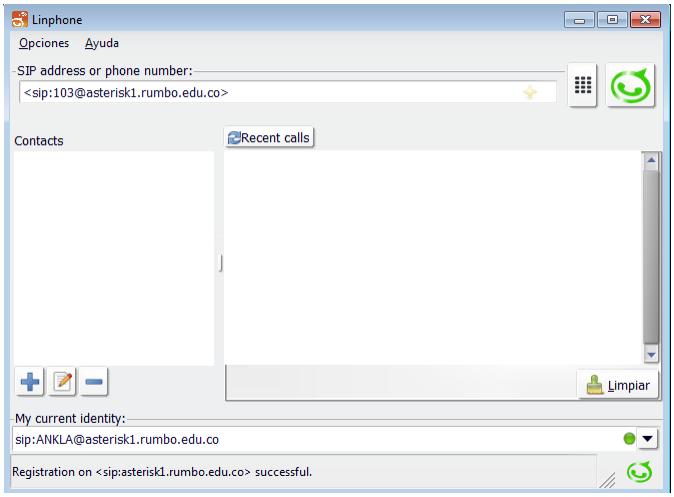
Ilustración 93. Parámetros de configuración para el registro de la cuenta SIP en el servidor Asterisk IPv6.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Se aceptan las configuraciones realizadas, se puede observar que el registro se realizó de forma satisfactoria, si aparece el letrero successful en la parte inferior.

Ilustración 94. Softphone registrado ante el servidor Asterisk IPv6.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

4.2.2 IMPLEMENTACIÓN SISTEMAS.

La implementación de los sistemas se realizan en dos fases, la primera fase se realiza la implementación de los protocolos MPLS y BGP en IPv4 y con el método del túnel para IPv6 de forma física en los routers Huawei que se encuentran en el laboratorio de la Universidad, la segunda fase se basa en la implementación de los mismo sistemas pero de manera emulada en el software GNS3.

CONFIGURACIÓN ROUTER HUAWEI SISTEMA MPLS IPv4 EXPERIEMNTO 1.

Esta configuración se implementa para desarrollar el experimento 1. Se ingresa al modo habilitado system-view en el router para iniciar la configuración, en primer lugar se asigna la dirección IP a las dos interfaces de cada router, también se asigna la dirección de Loopback 0, se crea el proceso de enrutamiento OSPF 10, asignando el área y las direcciones de red del router que está siendo implementado. Este proceso se realiza para todos los router de la estructura como se observa a continuación.

- Configuración de dirección IP en interfaces:

```
[R1] interface GigabitEthernet0/0/0
[R1] ip address 10.10.12.1 255.255.255.252
[R1] interface GigabitEthernet0/0/1
[R1] ip address 10.10.16.1 255.255.255.252
[R1] interface LoopBack0
[R1] ip address 1.1.1.1 255.255.255.255
```

- Configuración proceso OSPF:

```
[R1] ospf 10
[R1-ospf-10] area 0.0.0.0
[R1-ospf-10] 1.1.1.1 0.0.0.0
[R1-ospf-10] 10.10.12.0 0.0.0.3
[R1-ospf-10] 10.10.16.0 0.0.0.3
```

Este procedimiento se realiza en todos los 6 routers de la estructura diseñada anteriormente en la figura 17 y con las direcciones de la tabla 1, a continuación se configura el protocolo MPLS en todos los routers y por último se implementa la VPN entre los routers de borde.

Configuración routers de borde:

- Se ingresa al modo habilitado en los routers.
- Se habilita el protocolo MPLS con los comando mpls y mpls ldp, se ingresa a las interfaces para habilitar MPLS con los mismo comandos.
- Se habilita la vpn instance en los routers de borde (R5 y R6), habilitando los atributos de la vpn.
- Por último se implementa el protocolo BGP entre los routers de borde mediante la IP Loopback.

- Configuración de MPLS.

```
[R5] router id 50.50.50.50
[R5] mpls lsr-id 50.50.50.50
[R5] mpls
[R5] mpls ldp
[R5] interface GigabitEthernet0/0/0
[R5] mpls
[R5] mpls ldp
```

- Configuración de VPN instance.

```
[R5] ip vpn-instance tesis
[R5] ipv4-family
[R5] route-distinguisher 100:101
[R5] vpn-target 100:101 export-extcommunity
[R5] vpn-target 100:101 import-extcommunity 88
```

- Configuración de VPN en l interfaz.

```
[R5] interface GigabitEthernet0/0/1
[R5] ip binding vpn-instance tesis
[R5] ip address 192.168.0.1 255.255.255.0
```

- Configuración del protocolo BGP.

```
[R5] bgp 100
[R5] peer 60.60.60.60 as-number 100
[R5] peer 60.60.60.60 connect-interface LoopBack0
[R5] ipv4-family unicast
[R5] undo synchronization
```

[R5] undo peer 60.60.60.60 enable
[R5] ipv4-family vpn-instance tesis
[R5] import-route direct

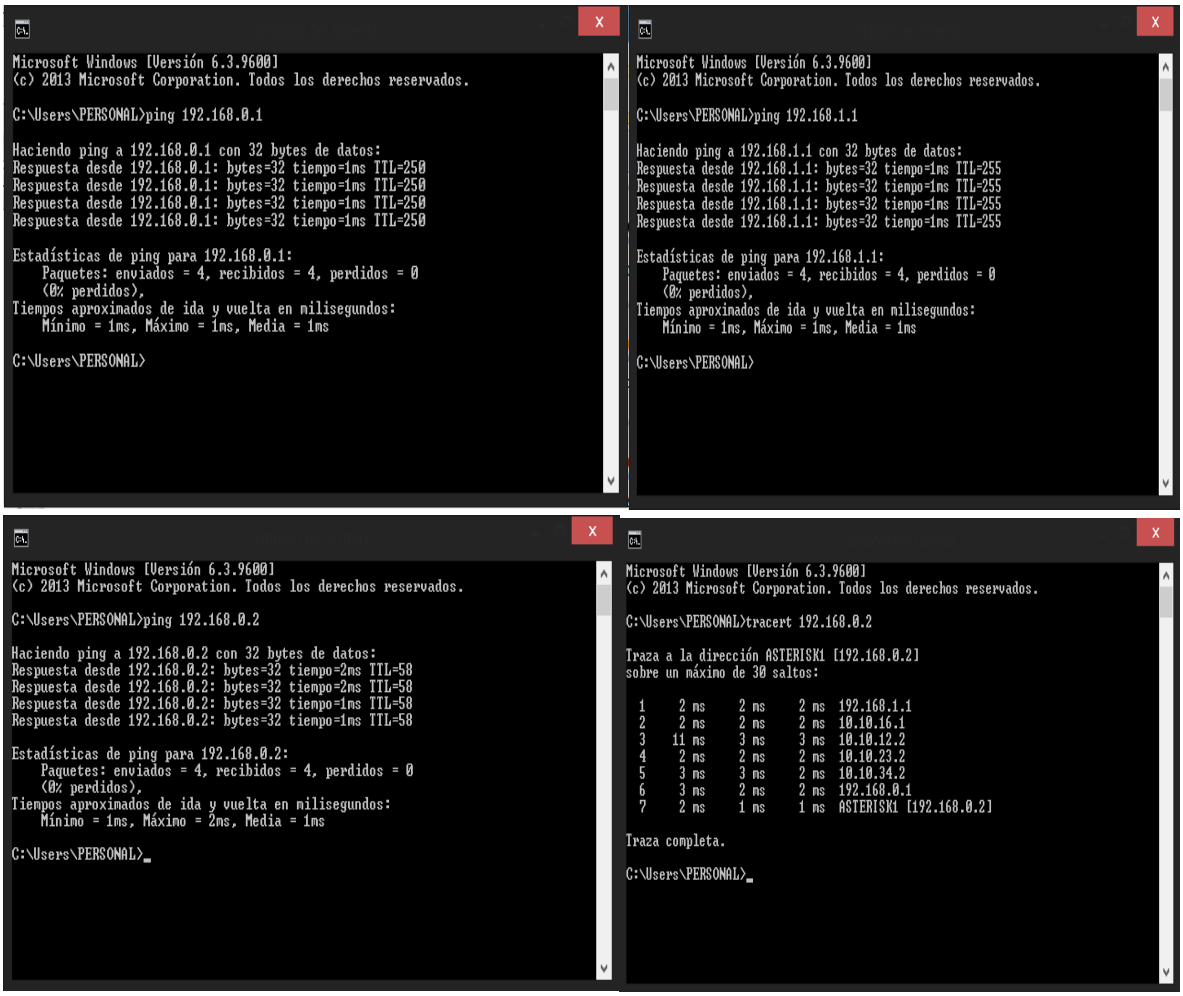
Teniendo configurado los routers de borde, se deben configurar los routers que se encuentran en el interior de la estructura (R1, R2, R3 y R4). Se implementa el protocolo MPLS en cada interfaz.

- Configuración de MPLS.

[R1] interface GigabitEthernet0/0/0
[R1] ip address 10.10.12.1 255.255.255.252
[R1] mpls
[R1] mpls ldp
[R1] interface GigabitEthernet0/0/1
[R1] ip address 10.10.16.1 255.255.255.252
[R1] mpls
[R1] mpls ldp

Prueba conectividad después de la configuración mediante ping desde el pc (softphone) hasta el servidor, router R5y R6 que están ubicados en el borde de la estructura.

Ilustración 95. Prueba conectividad.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

En la figura anterior se puede observar en la parte superior izquierda el ping desde el pc hasta el router R5, en la parte superior derecha el ping desde el pc hasta el router R6, en la parte inferior izquierda el ping desde el pc hasta el servidor Asterisk y por últimos en la parte inferior derecha un tracert desde el pc hasta el servidor Asterisk.

CONFIGURACIÓN ROUTER HUAWEI SISTEMA MPLS IPv6 EXPERIMENTO 2.

Esta configuración se implementa para desarrollar el experimento 2. La configuración para MPLS IPv6 se realiza sobre la base que se tiene para MPLS IPv4 ya que la implementación de este sistema busca un método de convivencia como lo es el túnel IPv6-IPv4. Se configura las direcciones IPv4 en todos los routers, se integra el proceso OSPF 10 en los routers R1, R2, R3 y R4 con excepción de la

red a la que pertenece la interfaz Gi 0/0/0 de los routers R1 y R4 donde se configura la VPN, se implementa el protocolo MPLS en los R1, R2, R3 y R4, por último se implementa el túnel entre el router R5 y el router R6 para brindar el servicio de voz en IPv6.

- Configuración de dirección IP en interfaces:

```
[R4] interface GigabitEthernet0/0/0
[R4] ip address 192.168.0.1 255.255.255.0
[R4] interface GigabitEthernet0/0/1
[R4] ip address 10.10.23.2 255.255.255.252
[R4] interface LoopBack0
[R4] ip address 50.50.50.50 255.255.255.255
```

- Configuración proceso OSPF:

```
[R4] ospf 10
[R4-ospf-10] area 0.0.0.0
[R4-ospf-10] 1.1.1.1 0.0.0.0
[R4-ospf-10] 10.10.12.0 0.0.0.3
[R4-ospf-10] 10.10.16.0 0.0.0.3
```

- Configuración de MPLS.

```
[R4] router id 50.50.50.50
[R4] mpls lsr-id 50.50.50.50
[R4] mpls
[R4] mpls ldp
[R4] interface GigabitEthernet0/0/1
[R4] mpls
[R4] mpls ldp
```

- Configuración de VPN instance.

```
[R4] ip vpn-instance tesis
[R4] ipv4-family
[R4] route-distinguisher 100:101
[R4] vpn-target 100:101 export-extcommunity
[R4] vpn-target 100:101 import-extcommunity 88
```

- Configuración de VPN en l interfaz.

```
[R4] interface GigabitEthernet0/0/1
[R4] ip binding vpn-instance tesis
[R4] ip address 192.168.0.1 255.255.255.0
```

- Configuración del protocolo BGP.

```
[R4] bgp 100
[R4] peer 60.60.60.60 as-number 100
[R4] peer 60.60.60.60 connect-interface LoopBack0
[R4] ipv4-family unicast
[R4] undo synchronization
[R4] undo peer 60.60.60.60 enable
[R4] ipv4-family vpn-instance tesis
[R4] import-route direct
```

A continuación se asigna la dirección IPv4 e IPv6 a los routers R5 y R6, también se implementa el túnel y por último se configura una ruta estática.

```
[R5] interface GigabitEthernet0/0/0
[R5] ip address 192.168.0.2 255.255.255.0
[R5] interface GigabitEthernet0/0/1
[R5] ipv6 enable
[R5] ip address 192.168.11.1 255.255.255.0
[R5] ipv6 address 2001:11::1/64
```

- Configuración túnel.

```
[R5] interface Tunnel0/0/0
[R5] ipv6 enable
[R5] ipv6 address 2001:10::6/64
[R5] tunnel-protocol ipv6-ipv4
[R5] source GigabitEthernet0/0/0
 [R5] destination 192.168.1.2
```

- Ruta estática.

```
[R5] ip route-static 192.168.1.0 255.255.255.0 192.168.0.1
[R5] ip route-static 192.168.20.0 255.255.255.0 192.168.0.1
```

Configuración R6

- Configuración túnel.

```
[R6] interface GigabitEthernet0/0/0
[R6] ip address 192.168.1.2 255.255.255.0
[R6] interface GigabitEthernet0/0/1
[R6] ipv6 enable
[R6] ip address 192.168.20.1 255.255.255.0
[R6] ipv6 address 2001:20::1/64
```

- Configuración túnel.

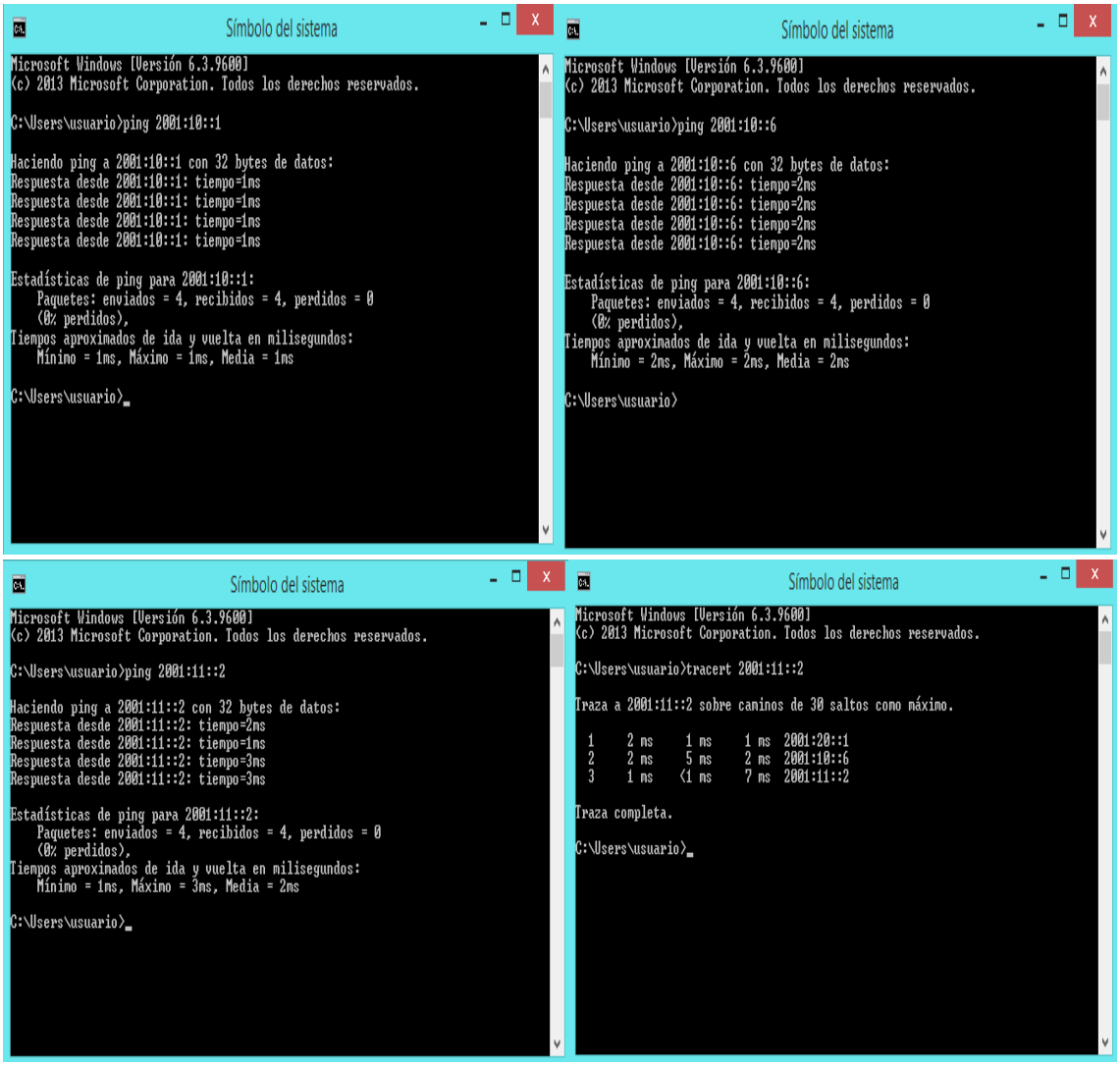
```
[R6] interface Tunnel0/0/0
[R6] ipv6 enable
[R6] ipv6 address 2001:10::1/64
[R6] tunnel-protocol ipv6-ipv4
[R6] source GigabitEthernet0/0/0
[R6] destination 192.168.0.2
```

- Ruta estática.

```
[R6] ip route-static 192.168.0.0 255.255.255.0 192.168.1.1
[R6] ip route-static 192.168.11.0 255.255.255.0 192.168.1.1
```

Prueba de conectividad después de la configuración mediante ping desde el pc (softphone) hasta el servidor, router R5y R6 que están ubicados en el borde de la estructura.

Ilustración 96. Prueba conectividad IPv6.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

En la figura anterior se puede observar en la parte superior izquierda el ping desde el pc hasta el túnel implementado en el router R5, en la parte superior derecha el ping desde el pc hasta el túnel implementado en el router R6, en la parte inferior izquierda el ping desde el pc hasta el servidor Asterisk y por últimos en la parte inferior derecha un tracert desde el pc hasta el servidor Asterisk.

CONFIGURACIÓN ROUTER HUAWEI SISTEMA BGP IPv4 EXPERIMENTO 3.

Esta configuración se implementa para desarrollar el experimento 3. Se ingresa al modo habilitado system-view en el router para iniciar la configuración, en primer lugar se asigna la dirección IP a las dos interfaces de cada router, también se asigna

la dirección de Loopback 0, se crea el proceso de enrutamiento OSPF 10, asignando el área y las direcciones de red del router que está siendo implementado. Este proceso se realiza para todos los router de la estructura como se observa a continuación.

- Configuración de dirección IP en interfaces:

```
[R1] interface GigabitEthernet0/0/0
[R1] ip address 10.10.3.2 255.255.255.252
[R1] interface GigabitEthernet0/0/1
[R1] ip address 10.10.4.1 255.255.255.252
[R1] interface LoopBack0
[R1] ip address 1.1.1.1 255.255.255.255
```

- Configuración proceso OSPF:

```
[R1] ospf 10
[R1-ospf-10] area 0.0.0.0
[R1-ospf-10] 1.1.1.1 0.0.0.0
[R1-ospf-10] 10.10.3.0 0.0.0.3
[R1-ospf-10] 10.10.4.0 0.0.0.3
```

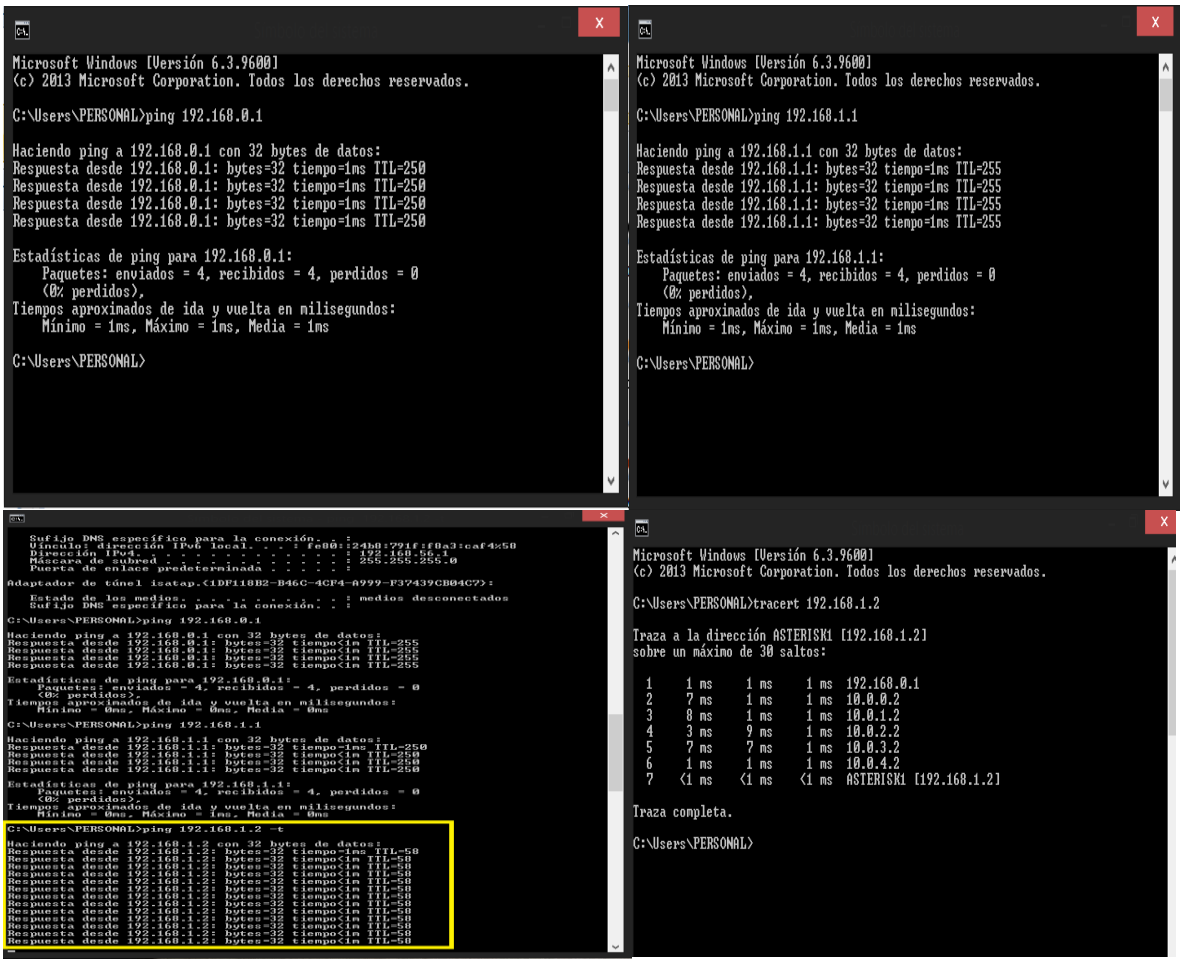
Este procedimiento se realiza en todos los 6 routers de la estructura diseñada anteriormente en la figura 19 y con las direcciones de la tabla 3, a continuación se configura el protocolo BGP en todos los routers, importando las rutas de todos los elementos de la estructura en cada router del sistema.

- Configuración BGP routers:

```
[R1] bgp 100
[R1-bgp-100] peer 2.2.2.2 as-number 100
[R1-bgp-100] peer 2.2.2.2 connect-interface Loopback0
[R1-bgp-100] peer 3.3.3.3 as-number 100
[R1-bgp-100] peer 3.3.3.3 connect-interface LoopBack0
[R1-bgp-100] peer 4.4.4.4 as-number 100
[R1-bgp-100] peer 4.4.4.4 connect-interface Loopback0
[R1-bgp-100] peer 50.50.50.50 as-number 100
[R1-bgp-100] peer 50.50.50.50 connect-interface LoopBack0
[R1-bgp-100]  peer 60.60.60.60 as-number 100
[R1-bgp-100]  peer 60.60.60.60 connect-interface Loopback0
```

Prueba de conectividad después de la configuración mediante ping desde el pc (softphone) hasta el servidor, router R5y R6 que están ubicados en el borde de la estructura.

Ilustración 97. Prueba conectividad.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014

En la figura anterior se puede observar en la parte superior izquierda el ping desde el pc hasta el router R5, en la parte superior derecha el ping desde el pc hasta el router R6, en la parte inferior izquierda el ping desde el pc hasta el servidor Asterisk y por últimos en la parte inferior derecha un tracert desde el pc hasta el servidor Asterisk.

CONFIGURACIÓN ROUTER HUAWEI SISTEMA BGP IPv6 EXPERIMENTO 4.

Esta configuración se implementa para desarrollar el experimento 4. La configuración para BGP IPv6 se realiza sobre la base que se tiene para BGP IPv4 ya que la implementación de este sistema busca un método de convivencia como lo es el túnel IPv6-IPv4.Se configura las direcciones IPv4 en todos los routers, se integra el proceso OSPF 10 en los routers en todos los routers con excepción de la red a la que pertenece la interfaz Gi 0/0/0 de los routers R5 y R6, se implementa el protocolo BGP en todos los routers para que importe la rutas de los elementos del sistema, por último se implementa el túnel entre el router R5 y el router R6 para brindar el servicio de voz en IPv6.

- Configuración de dirección IP en interfaces:

[R1] interface GigabitEthernet0/0/0
[R1] ip address 10.10.3.2 255.255.255.252
[R1] interface GigabitEthernet0/0/1
[R1] ip address 10.10.4.1 255.255.255.252
[R1] interface LoopBack0
[R1] ip address 1.1.1.1 255.255.255.255

- Configuración proceso OSPF:

[R1] ospf 10
[R1-ospf-10] area 0.0.0.0
[R1-ospf-10] 1.1.1.1 0.0.0.0
[R1-ospf-10] 10.10.3.0 0.0.0.3
[R1-ospf-10] 10.10.4.0 0.0.0.3

Este procedimiento se realiza en todos los 6 routers de la estructura diseñada anteriormente en la figura 20 y con las direcciones de la tabla 4, a continuación se configura el protocolo BGP en todos los routers, importando las rutas de todos los elementos de la estructura en cada router del sistema.

- Configuración BGP routers:

[R1] bgp 100
[R1-bgp-100] peer 2.2.2.2 as-number 100
[R1-bgp-100] peer 2.2.2.2 connect-interface Loopback0
[R1-bgp-100] peer 3.3.3.3 as-number 100
[R1-bgp-100] peer 3.3.3.3 connect-interface LoopBack0
[R1-bgp-100] peer 4.4.4.4 as-number 100

[R1-bgp-100] peer 4.4.4.4 connect-interface Loopback0
[R1-bgp-100] peer 50.50.50.50 as-number 100
[R1-bgp-100] peer 50.50.50.50 connect-interface LoopBack0
[R1-bgp-100] peer 60.60.60.60 as-number 100
[R1-bgp-100] peer 60.60.60.60 connect-interface Loopback0

A continuación se asigna la dirección IPv4 e IPv6 a los routers R5 y R6, también se implementa el túnel.

[R5] interface GigabitEthernet0/0/0
[R5] ip address 192.168.0.1 255.255.255.0
[R5] ipv6 enable
[R5] ip address 192.168.0.1 255.255.255.0
[R5] ipv6 address 2001:11::1/64
[R5] interface GigabitEthernet0/0/1
[R5] ip address 10.0.0.1 255.255.255.0

- Configuración túnel.

[R5] interface Tunnel0/0/1
[R5] ipv6 enable
[R5] ipv6 address 2001:10::1/64
[R5] tunnel-protocol ipv6-ipv4
[R5] source GigabitEthernet0/0/1
[R5] destination 10.0.4.2

- Ruta estática.

[R5] ipv6 route-static 2001:20:: 64 Tunnel0/0/1

Configuración R6

- Configuración túnel.

[R6] interface GigabitEthernet0/0/0
[R6] ip address 192.168.1.1 255.255.255.0
[R6] ipv6 enable
[R6] ipv6 address 2001:20::1/64
[R6] interface GigabitEthernet0/0/1

[R6] ip address 10.0.4.2 255.255.255.0

- Configuración túnel.

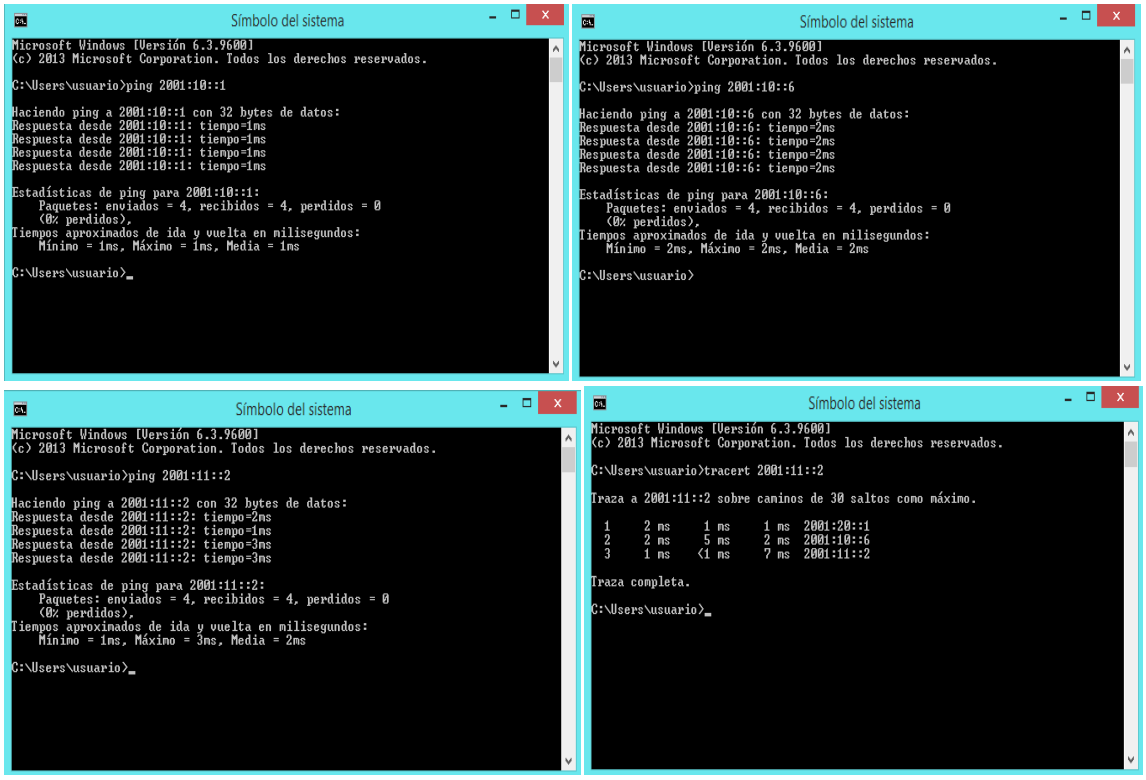
[R6] interface Tunnel0/0/1
[R6] ipv6 enable
[R6] ipv6 address 2001:10::6/64
[R6] tunnel-protocol ipv6-ipv4
[R6] source GigabitEthernet0/0/1
[R6] destination 10.0.0.1

- Ruta estática.

[R6] ipv6 route-static 2001:11:: 64 Tunnel0/0/1

Prueba de conectividad después de la configuración mediante ping desde el pc (softphone) hasta el servidor, router R5y R6 que están ubicados en el borde de la estructura.

Ilustración 98. Prueba conectividad IPv6.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

En la figura anterior se puede observar en la parte superior izquierda el ping desde el pc hasta el túnel implementado en el router R5, en la parte superior derecha el ping desde el pc hasta el túnel implementado en el router R6, en la parte inferior izquierda el ping desde el pc hasta el servidor Asterisk y por últimos en la parte inferior derecha un tracert desde el pc hasta el servidor Asterisk.

CONFIGURACIÓN SISTEMA MPLS IPv4 EMULADO (GNS3) EXPERIMENTO 5.

Esta configuración se implementa para desarrollar el experimento 5. Se ingresa al modo habilitado mediante el comando config terminal, se asigna el direccionamiento IPv4 que se encuentra en la tabla 5, se implementa el proceso OSPF 10 para el enrutamiento dinámico entre los routers, se habilita el protocolo mpls y mpls ldp, se asigna a las interfaces que van a funcionar con MPLS., este procedimiento se realiza en todos los routers de la estructura.

- Configuración interfaces.

```
R1# config terminal
R1(config) # interface GiEthernet 0/0
R1(config-if)# ip address 192.168.1.2 255.255.255.0
R1(config-if)# no shutdown
R1(config-if)# exit
R1(config) # interface GiEthernet 1/0
R1(config-if)# ip address 192.168.20.1 255.255.255.0
R1(config-if)# no shutdown
R1(config-if)# exit
R1(config)# interface loopback 0
R1(config-if)# ip address 10.10.10.1 255.255.255.255
R1(config-if)# exit
```

- Configuración OSPF 10.

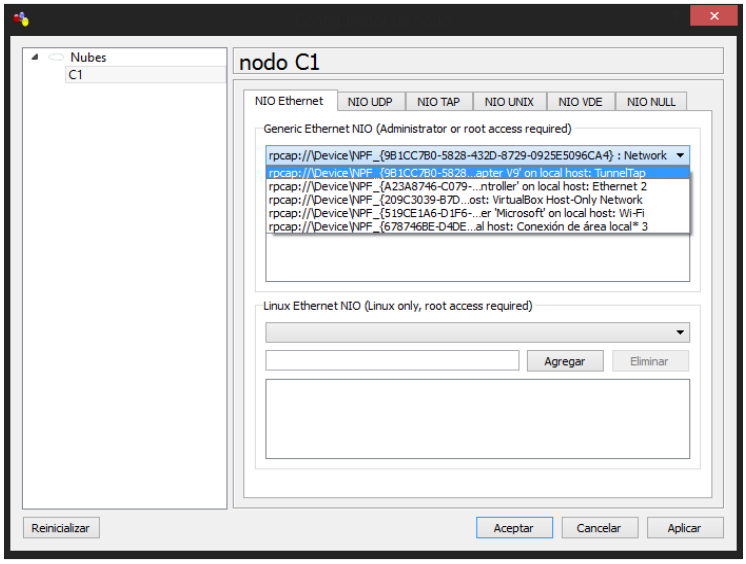
```
R1(config)# router ospf 10
R1(config-router)# router-id 10.10.10.1
R1(config-router)# network 10.10.10.1 0.0.0.0 area 0
R1(config-router)# network 192.168.1.2 0.0.0.0 area 0
```

- Configuración MPLS.

```
R1(config)# ip cef
R1(config)# mpls ip
R1(config)# mpls label protocol ldp
R1(config)# interface fastEthernet 0/0
R1(config-if)# mpls label protocol ldp
R1(config-if)# mpls ip
R1(config-if)# exit
R1(config)# interface fastEthernet 0/1
R1(config-if)# mpls label protocol ldp
R1(config-if)# mpls ip
R1(config-if)# exit
```

Las nubes se configuran dándole doble clic sobre la nube, se ingresa a la opción C1 y en la primera pestaña se abre el menú de interfaces disponibles seleccionando la que se le va asignar a esa nube. Para este caso se utilizan la interfaz identificada como Ethernet 2 y VirtualBox Host-Only Network. El proceso es igual para todas las implementaciones emuladas.

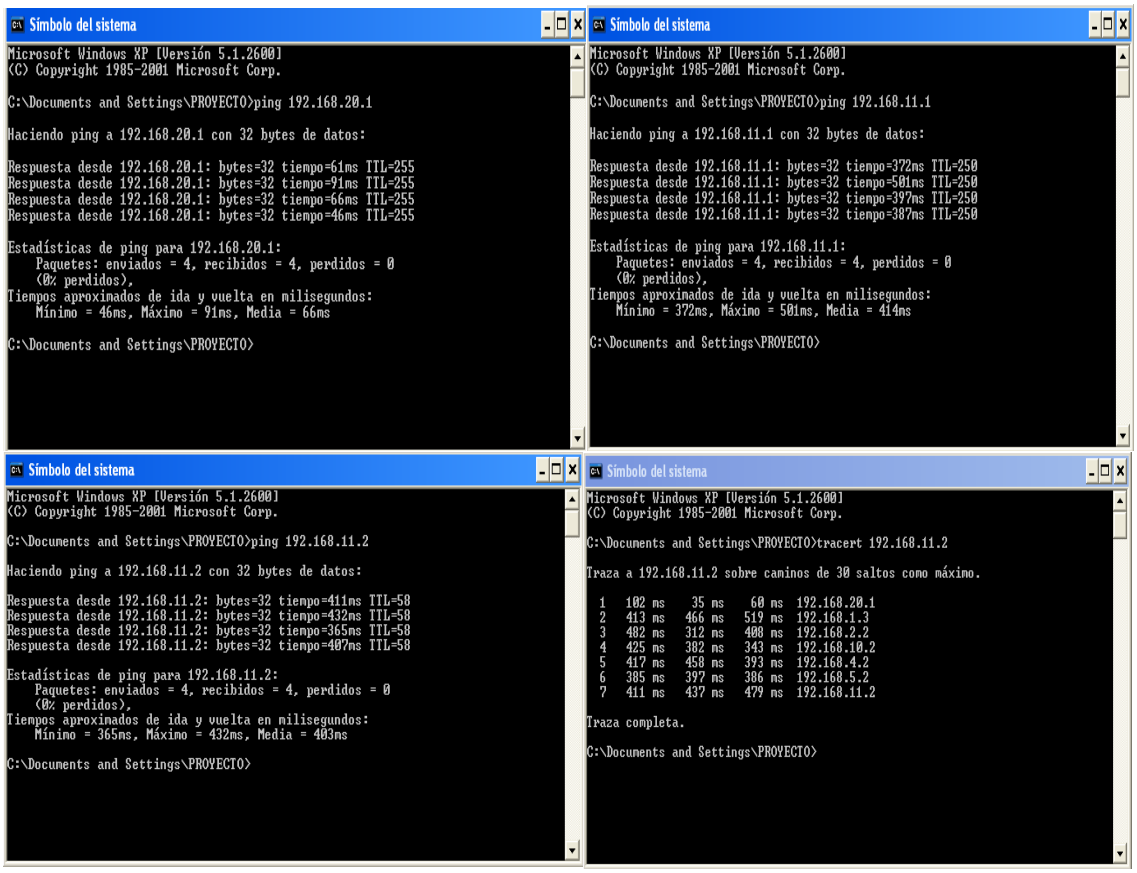
Ilustración 99. Configuración nubes GNs3



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Prueba conectividad después de la configuración mediante ping desde el pc (softphone) hasta el servidor, router R1 y R6 que están ubicados en el borde de la estructura.

Ilustración 100. Prueba conectividad.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014

En la figura anterior se puede observar en la parte superior izquierda el ping desde el pc hasta el router R1, en la parte superior derecha el ping desde el pc hasta el router R6, en la parte inferior izquierda el ping desde el pc hasta el servidor Asterisk y por últimos en la parte inferior derecha un tracert desde el pc hasta el servidor Asterisk.

CONFIGURACIÓN SISTEMA MPLS IPv6 EMULADO (GNS3) EXPERIMENTO 6.

Esta configuración se implementa para desarrollar el experimento 6. La configuración para MPLS IPv6 se realiza sobre la base que se tiene para MPLS IPv4 ya que la implementación de este sistema busca un método de convivencia como lo es el túnel IPv6-IPv4.Se configura las direcciones IPv4 en todos los routers, se integra el proceso OSPF 10 en los routers de la estructura, se implementa el protocolo MPLS, por último se implementa el túnel entre el router R1 y el router R6 para brindar el servicio de voz en IPv6.

- Configuración interfaces.

```
R1# config terminal
R1(config) # interface GiEthernet 0/0
R1(config-if)# ip address 192.168.1.2 255.255.255.0
R1(config-if)# no shutdown
R1(config-if)# exit
R1(config) # interface GiEthernet 1/0
R1(config-if)# ip address 192.168.20.1 255.255.255.0
R1(config-if)# no shutdown
R1(config-if)# exit
R1(config) # interface GiEthernet 1/0
R1(config-if)# ipv6 address 2001:20::1/64
R1(config-if)# exit
R1(config)# interface loopback 0
R1(config-if)# ip address 10.10.10.1 255.255.255.255
R1(config-if)# exit
R1(config)# interface Loopback1
R1(config-if)#  ipv6 address 2001:1::1/64
```

- Configuración OSPF 10.

```
R1(config)# router ospf 10
R1(config-router)# router-id 10.10.10.1
R1(config-router)# network 10.10.10.1 0.0.0.0 area 0
R1(config-router)# network 192.168.1.2 0.0.0.0 area 0
```

- Configuración MPLS.

```
R1(config)# ip cef
R1(config)# mpls ip
R1(config)# mpls label protocol ldp

R1(config)# interface fastEthernet 0/0
R1(config-if)# mpls label protocol ldp
R1(config-if)# mpls ip
R1(config-if)# exit

R1(config)# interface fastEthernet 0/1
R1(config-if)# mpls label protocol ldp
```

```
R1(config-if)# mpls ip
R1(config-if)# exit
```

- Configuración túnel IPv6-IPv4.

```
R1(config)# interface Tunnel0
R1(config-if)# ipv6 address 2001:10::1/64
R1(config-if)# tunnel source FastEthernet0/0
R1(config-if)#  tunnel destination 192.168.5.2
R1(config-if)# tunnel mode ipv6-ipv4
```

- Ruta estática.

```
R1(config)# ipv6 route 2001:11::/64 2001:6::6
```

Configuración R6.

- Configuración túnel IPv6-IPv4.

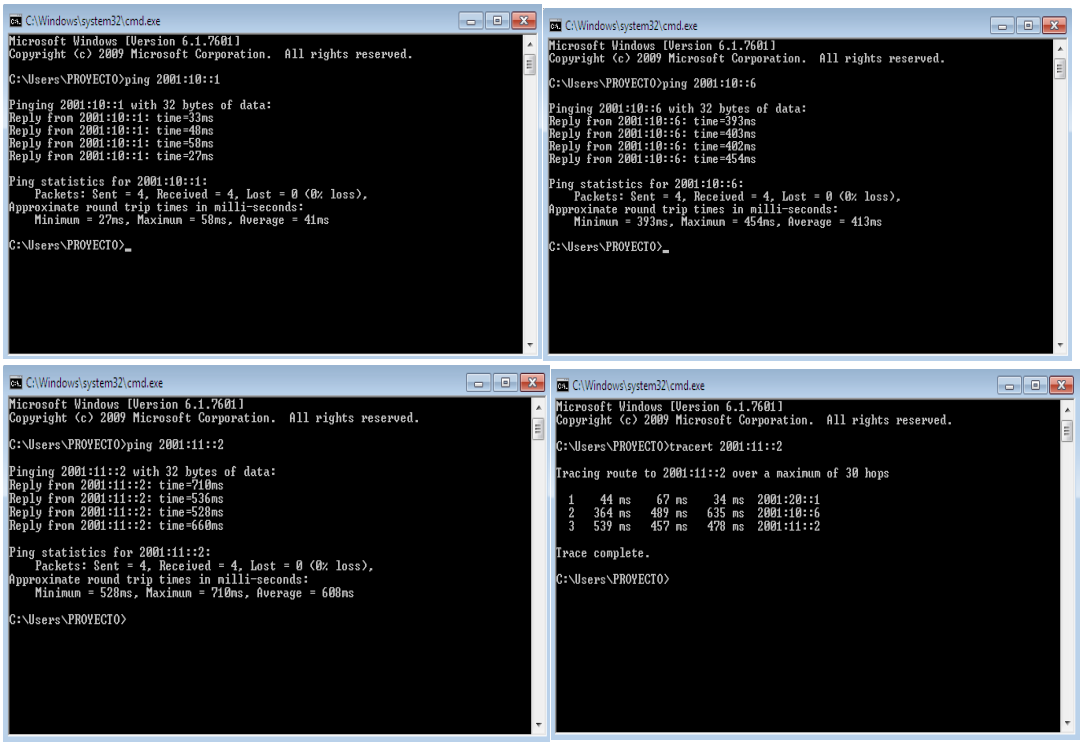
```
R6(config)# interface Tunnel0
R6(config-if)# ipv6 address 2001:10::6/64
R6(config-if)# tunnel source FastEthernet0/0
R6(config-if)#  tunnel destination 192.168.1.2
R6(config-if)# tunnel mode ipv6-ipv4
```

- Ruta estática.

```
R1(config)# ipv6 route 2001:20::/64 2001:1::1
```

Prueba de conectividad después de la configuración mediante ping desde el pc (softphone) hasta el servidor, router R1 y R6 que están ubicados en el borde de la estructura.

Ilustración 101. Prueba conectividad IPv6.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

En la figura anterior se puede observar en la parte superior izquierda el ping desde el pc hasta el túnel implementado en el router R1, en la parte superior derecha el ping desde el pc hasta el túnel implementado en el router R6, en la parte inferior izquierda el ping desde el pc hasta el servidor Asterisk y por últimos en la parte inferior derecha un tracert desde el pc hasta el servidor Asterisk.

CONFIGURACIÓN SISTEMA BGP IPv4 EMULADO (GNS3) EXPERIMENTO 7.

Esta configuración se implementa para desarrollar el experimento 7. Se ingresa al modo habilitado mediante el comando config terminal, en primer lugar se asigna la dirección IP a las dos interfaces de cada router, también se asigna la dirección de Loopback 0, se crea el proceso de enrutamiento OSPF 10, asignando el área y las direcciones de red del router que está siendo implementado. Este proceso se realiza para todos los router de la estructura como se observa a continuación.

- Configuración interfaces.

R1# config terminal
R1(config) # interface GiEthernet 0/0

```
R1(config-if)# ip address 10.0.0.1 255.255.255.0
R1(config-if)# no shutdown
R1(config-if)# exit
R1(config) # interface GiEthernet 1/0
R1(config-if)# ip address 192.168.20.1 255.255.255.0
R1(config-if)# no shutdown
R1(config-if)# exit
R1(config)# interface loopback 0
R1(config-if)# ip address 1.1.1.1 255.255.255.255
R1(config-if)# exit
R1(config)# interface Loopback1
R1(config-if)# ip address 11.11.11.11 255.255.255.255
R1(config-if)# exit
```

- Configuración OSPF 10.

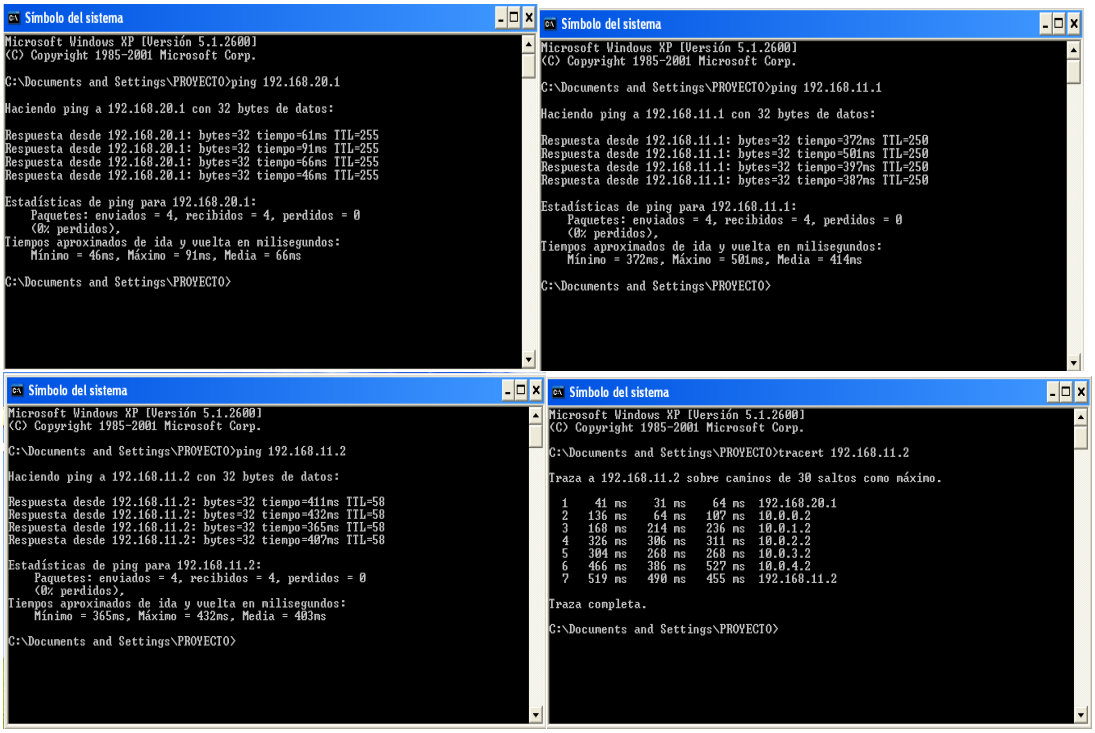
```
R1(config)# router ospf 10
R1(config-router)# router-id 1.1.1.1
R1(config-router)# network 1.1.1.1 0.0.0.0 area 0
R1(config-router)# network 10.0.0.0 0.0.0.255 area 0
```

- Configuración BGP.

```
R1(config)# router bgp 100
R1(config-router)# network 11.11.11.0 mask 255.255.255.0
R1(config-router)# neighbor 2.2.2.2 remote-as 100
R1(config-router)# neighbor 2.2.2.2 update-source Loopback0
R1(config-router)# neighbor 3.3.3.3 remote-as 100
R1(config-router)# neighbor 3.3.3.3 update-source Loopback0
R1(config-router)# neighbor 4.4.4.4 remote-as 100
R1(config-router)# neighbor 4.4.4.4 update-source Loopback0
R1(config-router)# neighbor 5.5.5.5 remote-as 100
R1(config-router)# neighbor 5.5.5.5 update-source Loopback0
R1(config-router)# neighbor 6.6.6.6 remote-as 100
R1(config-router)# neighbor 6.6.6.6 update-source Loopback0
```

Prueba de conectividad después de la configuración mediante ping desde el pc (softphone) hasta el servidor, router R1 y R6 que están ubicados en el borde de la estructura.

Ilustración 102. Prueba conectividad.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014

En la figura anterior se puede observar en la parte superior izquierda el ping desde el pc hasta el router R1, en la parte superior derecha el ping desde el pc hasta el router R6, en la parte inferior izquierda el ping desde el pc hasta el servidor Asterisk y por últimos en la parte inferior derecha un tracert desde el pc hasta el servidor Asterisk.

CONFIGURACIÓN SISTEMA BGP IPv6 EMULADO (GNS3) EXPERIMENTO 8.

Esta configuración se implementa para desarrollar el experimento 8. La configuración para BGP IPv6 se realiza sobre la base que se tiene para BGP IPv4 ya que la implementación de este sistema busca un método de convivencia como lo es el túnel IPv6-IPv4.Se configura las direcciones IPv4 en todos los routers, se integra el proceso OSPF 10 en los routers en todos los routers con excepción de la red a la que pertenece la interfaz Gi 0/0/0 de los routers R1 y R6, se implementa el protocolo BGP en rodos lo router para que importe la rutas de los elementos del sistema, por último se implementa el túnel entre el router R1 y el router R6 para brindar el servicio de voz en IPv6.

- Configuración interfaces.

```
R1# config terminal
R1(config) # interface GiEthernet 0/0
R1(config-if)# ip address 10.0.0.1 255.255.255.0
R1(config-if)# no shutdown
R1(config-if)# exit
R1(config) # interface GiEthernet 1/0
R1(config-if)# ip address 192.168.20.1 255.255.255.0
R1(config-if)# no shutdown
R1(config-if)# exit
R1(config) # interface GiEthernet 1/0
R1(config-if)# ipv6 address 2001:20::1/64
R1(config-if)# exit
R1(config)# interface loopback 0
R1(config-if)# ip address 1.1.1.1 255.255.255.255
R1(config-if)# exit
R1(config)# interface Loopback1
R1(config-if)# ip address 11.11.11.11 255.255.255.255
R1(config-if)# exit
R1(config)# interface Loopback2
R1(config-if)#  ipv6 address 2001:1::1/64
```

- Configuración OSPF 10.

```
R1(config)# router ospf 10
R1(config-router)# router-id 1.1.1.1
R1(config-router)# network 1.1.1.1 0.0.0.0 area 0
R1(config-router)# network 10.0.0.0 0.0.0.255 area 0
```

- Configuración BGP.

```
R1(config)# router bgp 100
R1(config-router)# network 11.11.11.0 mask 255.255.255.0
R1(config-router)# neighbor 2.2.2.2 remote-as 100
R1(config-router)# neighbor 2.2.2.2 update-source Loopback0
R1(config-router)# neighbor 3.3.3.3 remote-as 100
R1(config-router)# neighbor 3.3.3.3 update-source Loopback0
```

```
R1(config-router)# neighbor 4.4.4.4 remote-as 100
R1(config-router)# neighbor 4.4.4.4 update-source Loopback0
R1(config-router)# neighbor 5.5.5.5 remote-as 100
R1(config-router)# neighbor 5.5.5.5 update-source Loopback0
R1(config-router)# neighbor 6.6.6.6 remote-as 100
R1(config-router)# neighbor 6.6.6.6 update-source Loopback0
```

- Configuración túnel IPv6-IPv4.

```
R1(config)# interface Tunnel0
R1(config-if)# ipv6 address 2001:10::1/64
R1(config-if)# tunnel source FastEthernet0/0
R1(config-if)#  tunnel destination 192.168.5.2
R1(config-if)# tunnel mode ipv6-ipv4
```

- Ruta estática.

```
R1(config)# ipv6 route 2001:11::/64 2001:6::6
```

Configuración R6.

- Configuración túnel IPv6-IPv4.

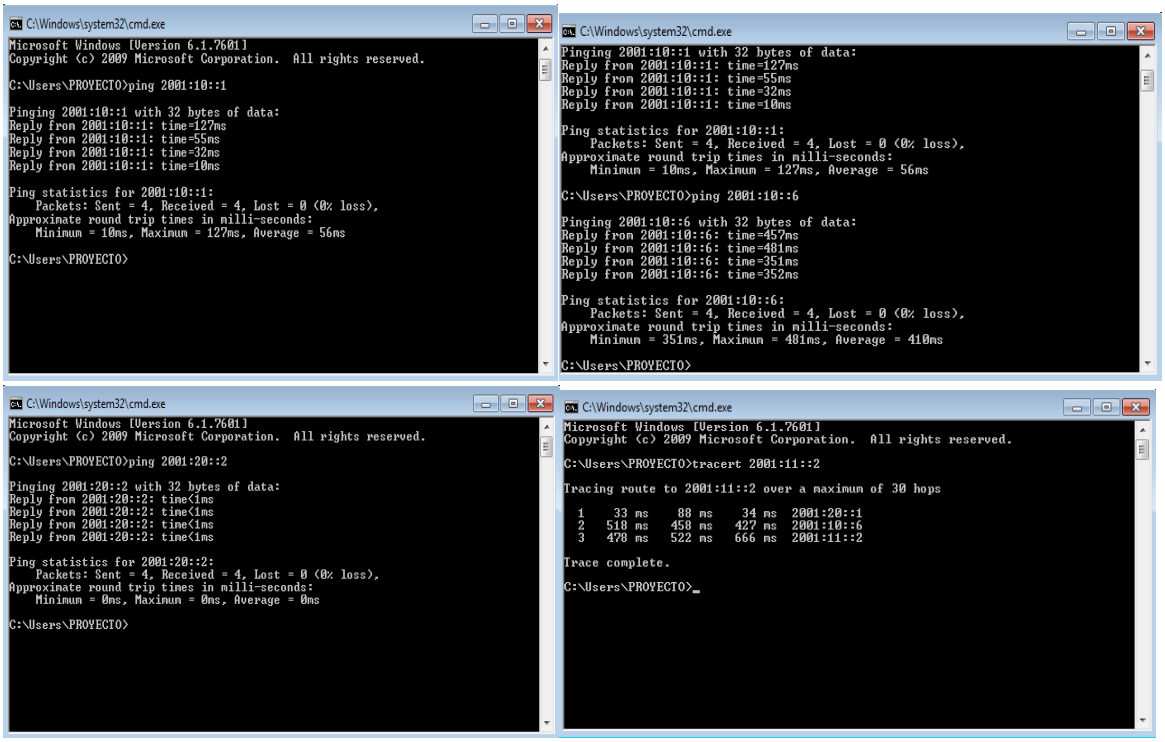
```
R6(config)# interface Tunnel0
R6(config-if)# ipv6 address 2001:10::6/64
R6(config-if)# tunnel source FastEthernet0/0
R6(config-if)#  tunnel destination 192.168.1.2
R6(config-if)# tunnel mode ipv6-ipv4
```

- Ruta estática.

```
R1(config)# ipv6 route 2001:20::/64 2001:1::1
```

Prueba de conectividad después de la configuración mediante ping desde el pc (softphone) hasta el servidor, router R1 y R6 que están ubicados en el borde de la estructura.

Ilustración 103. Prueba conectividad IPv6.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

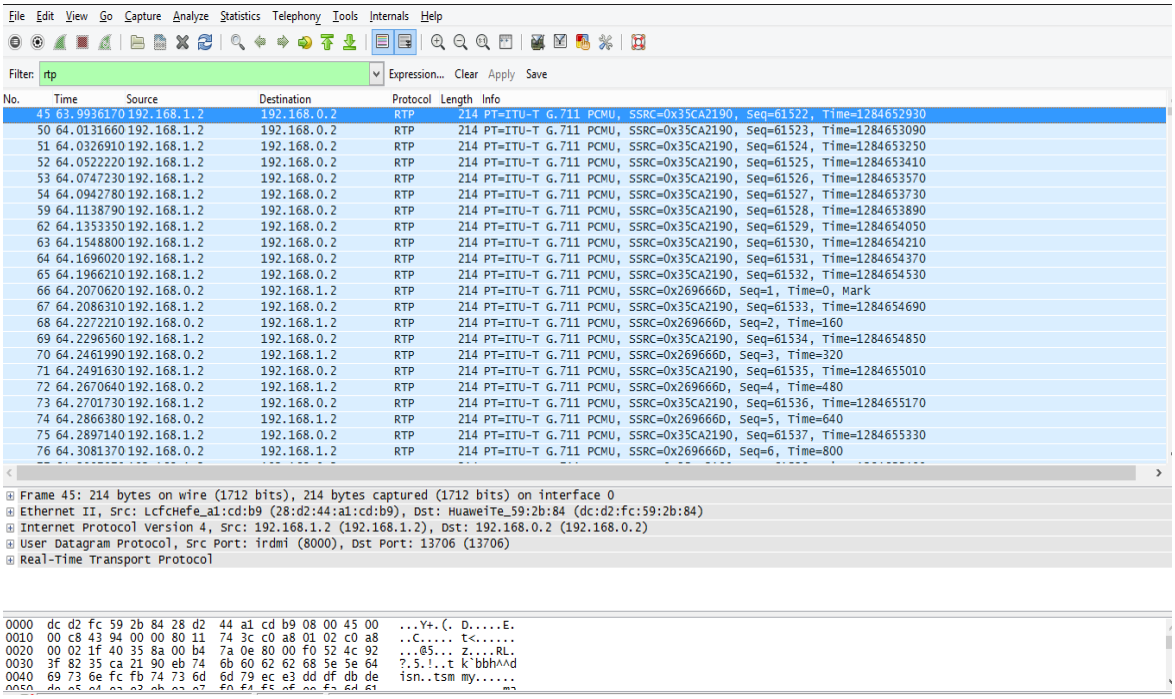
En la figura anterior se puede observar en la parte superior izquierda el ping desde el pc hasta el túnel implementado en el router R1, en la parte superior derecha el ping desde el pc hasta el túnel implementado en el router R6, en la parte inferior izquierda el ping desde el pc hasta el servidor Asterisk y por últimos en la parte inferior derecha un tracert desde el pc hasta el servidor Asterisk

5. RESULTADOS DE EXPERIMENTOS

Los datos de los resultados se obtuvieron mediante capturas de los paquetes RTP transmitidos por los sistemas MPLS y BGP anteriormente configurados mediante el software Wireshark, esta herramienta brinda la funcionalidad de filtrar los datos específicamente por lo que se necesite. El jitter y delay son valores promedios obtenidos de la captura de 25 minutos de los paquetes RTP.

En cada uno de los experimento se realizó la captura de los paquetes RPT, a continuación se observa uno de los casos.

Ilustración 104. Captura paquetes RTP.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

A continuación se exportan los datos a Excel para su posterior análisis y poder obtener el jitter y delay de cada sistema implementado.

5.1 RESULTADOS EXPERIMENTOS REALES.

Los datos que se exportan a Excel son el jitter y el delay por paquete, por lo que se toman los datos y se calcula el promedio para poder medir el jitter y delay total. En

la siguiente tabla se observa los resultados para los sistemas implementado en los routers Huawei.

Tabla 11. Resultados experimentos reales.

Variables\sistemas	MPLS IPv4	MPLS IPv6	BGP IPV4	BGP IPV6
JITTER	0,559320235 ms	19,9318659 ms	6,29383764 ms	20,0127247 ms
DELAY	19,9995574 ms	20,0367159 ms	19,9999807 ms	20,9366278 ms

Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

5.2 RESULTADOS EXPERIMENTOS EMULADOS.

Los datos que se obtienen en la captura son el jitter y el delay por paquete, por lo que se deben tomar los datos y sacar el promedio para poder calcular el jitter y delay final. En la siguiente tabla se observa los resultados para lo experimento realizados en el emular GNS3.

Tabla 12. Resultados experimentos emulados.

Variables\sistemas	MPLS IPv4	MPLS IPv6	BGP IPV4	BGP IPV6
JITTER	46,3087496 ms	60,5731066 ms	61,7153519 ms	72,1560436 ms
DELAY	33,5745459 ms	36,5334404 ms	33,5668641 ms	38,80898 ms

Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

6. ANÁLISIS DE RESULTADOS

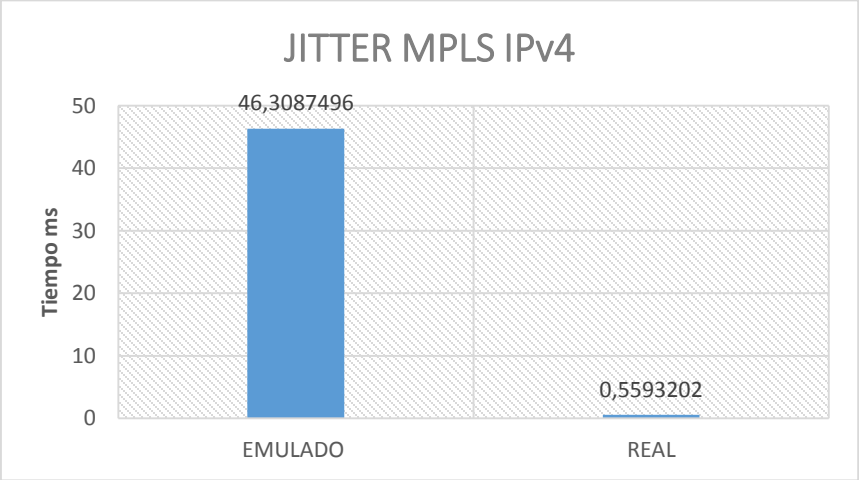
Obteniendo los resultados de los experimentos se procede a realizar el correspondiente análisis de cada uno de los datos de los sistemas implementados tanto para IPv4 como para el método de convivencia IPv6, donde se observa la diferencia del rendimiento que presenta el servicio de voz IP.

El análisis se realiza comparando los experimentos entre sí observando las diferencias.

6.1 ANÁLISIS JITTER Y DELAY MPLS IPV4.

Se observa a continuación el análisis que se le realizó a la prueba de llamada con duración de 25 minutos sobre el sistema MPLS IPv4 tanto de forma real en los routers Huawei como de forma emulada en el software GNs3.

Ilustración 105. Jitter MPLS IPv4.

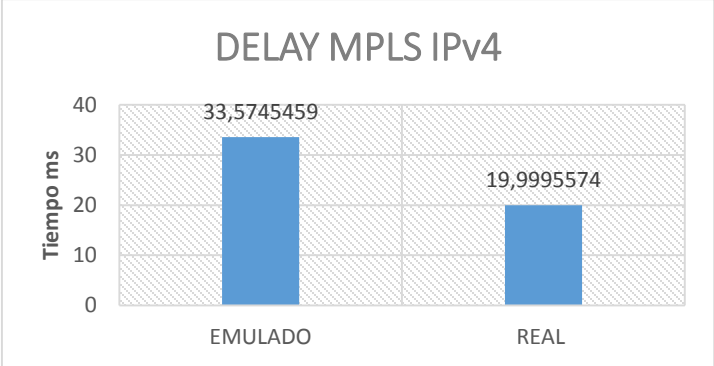


Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

En la figura anterior se puede evidenciar que el jitter en el sistema emulado es mucho mayor que en el sistema implementado en los routers Huawei, demostrando así los resultados que se esperaban teóricamente (Daisaku Shimazaki, october 2008). De igual manera la llamada realizada en el sistema emulado se pudo realizar sin inconvenientes durante los 25 minutos propuestos, con una calidad de Voz baja posiblemente debido al alto jitter de la prueba en GNs3.

Para las pruebas realizas, los resultados de Delay que se obtuvieron en el sistema MPLS IPv4 de forma emulado y real son los siguientes:

Ilustración 106. Delay MPLS IPv4.



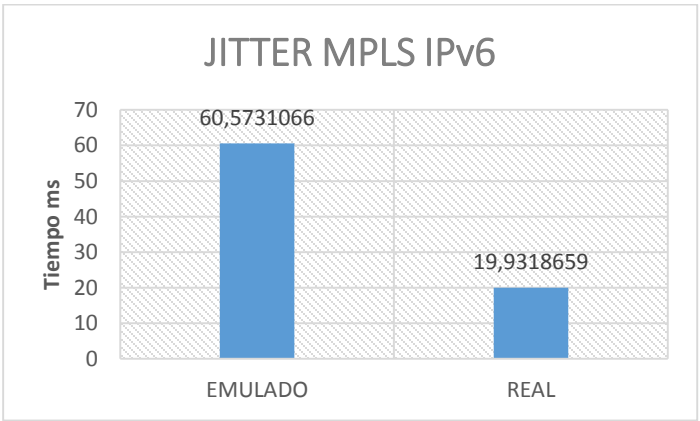
Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Se observa que en la variable Delay se obtiene el mismo comportamiento que para el Jitter donde los valores de las variables en el sistema emulado son mucho mayores que en el sistema real, esto es posiblemente debido a que los routers Huawei cuentan con un procesamiento más rápido que el del emulador GNs3.

6.2 ANÁLISIS JITTER Y DELAY MPLS IPv6.

Se observa a continuación el análisis que se le realizo a la prueba de Voz con duración de 25 minutos sobre el sistema MPLS IPv6, implementando el método de convivencia de túnel IPv6-IPv4, tanto de forma real en los routers Huawei como de forma emulada en el software GNs3.

Ilustración 107. Jitter MPLS IPv6.

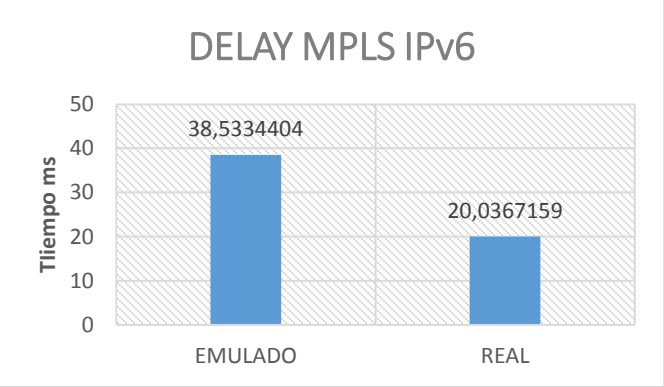


Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

En la figura anterior se puede evidenciar que el jitter en el sistema emulado es mucho mayor que en el sistema implementado en los routers Huawei, por tanto con base en estos resultados se puede decir que el método de convivencia del túnel IPv6-IPv4 no es recomendable para realizar implementaciones voz sobre IPv6 ya que muestra altos valores de jitter y delay. Para la llamada realizada en el sistema MPLS emulado se presentan inconvenientes ya que la llamada presentá caídas en un promedio de 18 a 20 minutos, también se notó que la calidad es muy baja y se tuvieron pérdidas de paquetes.
Este problema posiblemente se presenta debido al túnel implementado y por el equipo que se utiliza para el desarrollo de la prueba emulada.

Para las pruebas realizas, los resultados de Delay que se obtuvieron en el sistema MPLS IPv6 de forma emulado y real son los siguientes:

Ilustración 108. Delay MPLS IPv6.



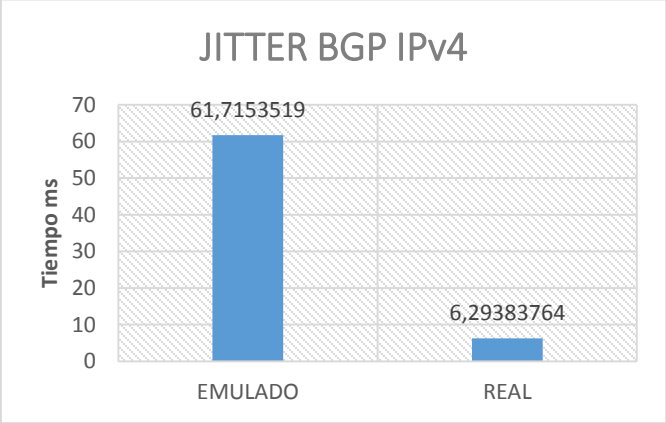
Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Se observa que en la variable Delay se obtiene muy similar al de Jitter la prueba anterior donde el valor de la variable en el sistema emulado es mucho mayor que en el sistema real. Debido a que los routers Huawei cuenta con un procesamiento más rápido que el del emulador GNs3. Se aclara que si el método de convivencia hubiera sido dual stack posiblemente los resultados que se obtendrían serian diferentes a los obtenidos de este proyecto.

6.3 ANÁLISIS JITTER Y DELAY BGP IPv4.

Se observa a continuación el análisis que se le realizo a la prueba de llamada con duración de 25 minutos sobre el sistema BGP IPv4 tanto de forma real en los routers Huawei como de forma emulada en el software GNs3.

Ilustración 109. Jitter BGP IPv4.

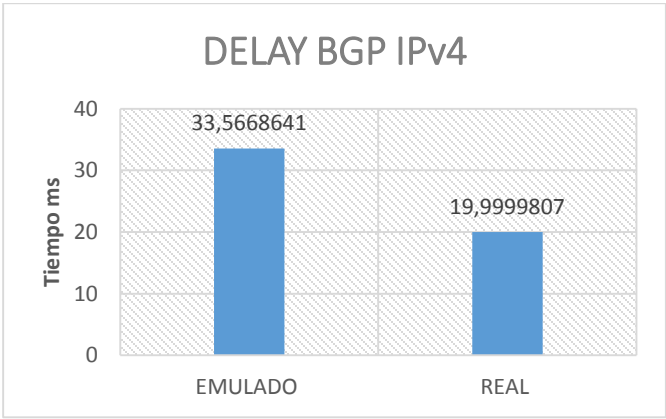


Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

En la figura anterior se puede evidenciar que el jitter en el sistema emulado es mucho mayor que en el sistema implementado en los routers Huawei, demostrando así los resultados que se esperaban teóricamente. De igual manera la llamada realizada en el sistema emulado se pudo realizar sin inconvenientes durante los 25 minutos propuestos, debido a los efectos del alto jitter y dealy la llamada tenía una calidad baja donde la voz que se emitía desde el transmisor a destino se escuchaba entrecortada.

Para las pruebas realizas, los resultados de Delay que se obtuvieron en el sistema BGP IPv4 de forma emulado y real son los siguientes:

Ilustración 110. Delay BGP IPv4



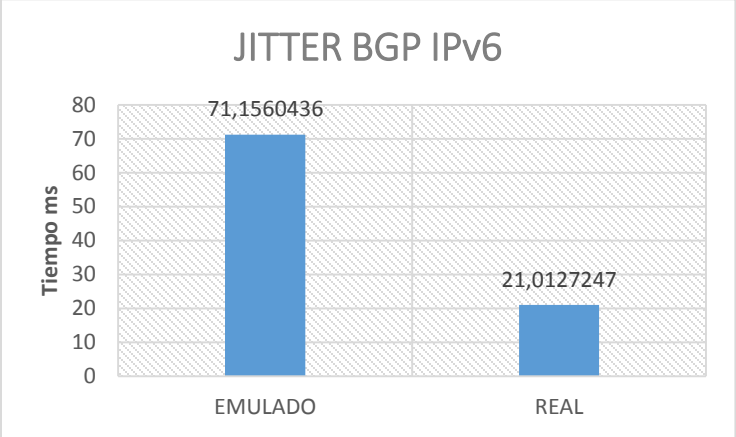
Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Al igual que en los experimentos anteriores, se observa que en la variable Delay tiene el mismo comportamiento que la variable Jitter en donde los valores de las mismas son más elevados en las pruebas realizada en el emulador.

6.4 ANÁLISIS JITTER Y DELAY BGP IPv6.

Se observa a continuación el análisis que se le realizo a la prueba de llamada con duración de 25 minutos sobre el sistema MPLS IPv6, implementando el método de convivencia de túnel IPv6-IPv4, tanto de forma real en los routers Huawei como de forma emulada en el software GNs3.

Ilustración 111. Jitter BGP IPv6.



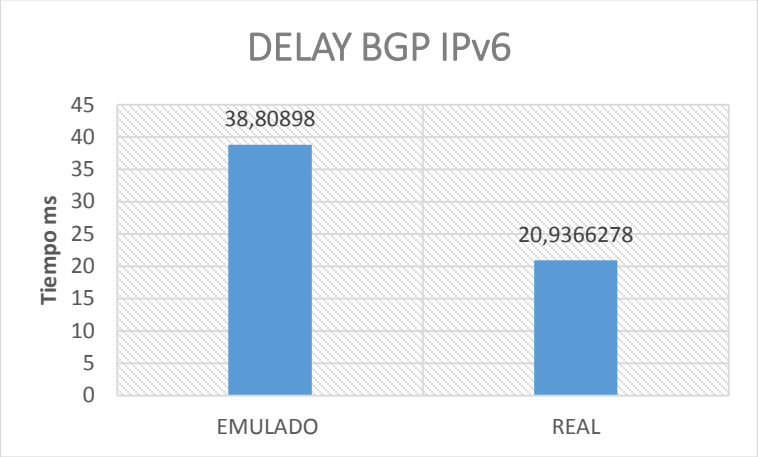
Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

En la figura anterior se puede evidenciar que el jitter en el sistema emulado es mayor que en el sistema implementado en los routers Huawei, lo cual tiene consecuencias negativas en la calidad de la voz.

De igual manera la llamada realizada en el sistema BGP real no presentó ningún inconveniente durante los 25 minutos propuestos, con una buena calidad y sin interferencias, para la llamada realizada en el sistemas BGP emulado se presentan inconvenientes ya que la llamada tiene un promedio de 11 a 13 minutos antes que se corte, la calidad es muy baja con pérdida de información. El problema se presenta debido al túnel implementado y por el equipo que se utiliza para el desarrollo de la prueba emulada.

Para las pruebas realizas, los resultados de Delay que se obtuvieron en el sistema BGP IPv6 de forma emulado y real son los siguientes:

Ilustración 112. Delay BGP IPv6.



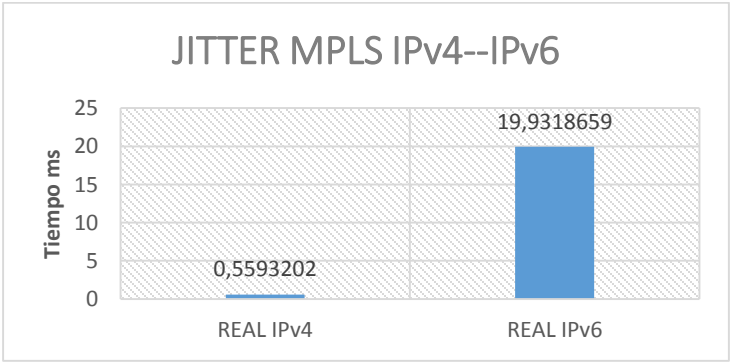
Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Se observa que en la variable Delay se obtiene el mismo resultado que para el Jitter donde el tiempo en el sistema emulado es mucho mayor que en el sistema real. Debido a que los routers Huawei cuenta con un procesamiento más rápido que el del emulador GNs3. Se aclara que si el método de convivencia hubiera sido diferente como dual stack los resultados que se obtendrían serian diferentes a los de este proyecto.

6.5 ANÁLISIS JITTER Y DELAY MPLS IPV4 Vs. IPv6 REAL.

Se observa a continuación el análisis comparativo entre el sistema MPLS IPv4 y el sistema MPLS IPv6, implementado en los routers Huawei. Para IPv6 se utiliza el método de convivencia de túnel IPv6-IPv4. Las pruebas se llevaron a cabo durante 25 minutos.

Ilustración 113. Jitter MPLS IPv4-IPv6 real.

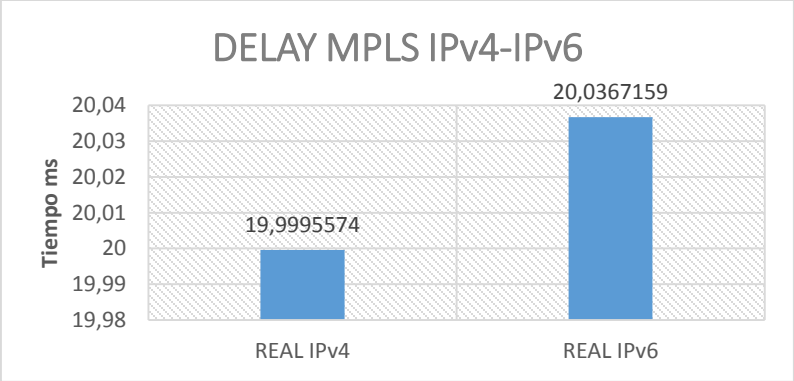


Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

En la figura anterior se puede evidenciar que el jitter en el sistema MPLS IPv6 es mucho mayor que en el sistema MPLS IPv4 lo que tiene consecuencias negativas en la calidad de la Voz esto posiblemente se deba al método de convivencia utilizado para realizar las pruebas (túnel IPv6-IPv4).

Para las pruebas realizas, los resultados de Delay que se obtuvieron en los sistemas MPLS IPv4 y MPLS IPv6 de forma real son los siguientes:

Ilustración 114. Delay MPLS IPv4-IPv6 real.



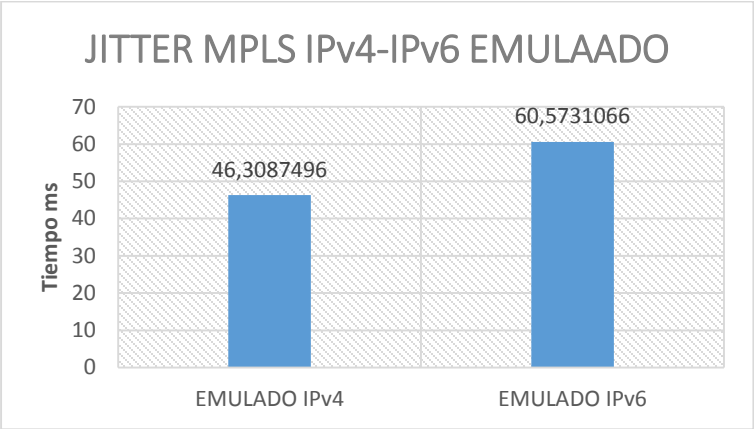
Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

En esta gráfica no se muestra una gran variación en el Delay al usar el método de convivencia, por tanto según los efectos percibidos a realizar los experimentos en la calidad de voz, se puede decir que el Jitter tiene un efecto negativo más relevante que el Delay y que usar el túnel IPv6-IPv4 como método de convivencia tiene efectos negativos en las comunicaciones de Voz sobre IPv6 pues aumenta en gran medida el Jitter en una de la red. De igual manera se aclara que si el método de convivencia hubiera sido dual stack los resultados que se obtendrían serian diferentes a los de este proyecto.

6.6 ANÁLISIS JITTER Y DELAY MPLS IPV4 Vs. IPv6 EMULADO.

Se observa a continuación el análisis comparativo entre el sistema MPLS IPv4 y el sistema MPLS IPv6, implementado de manera emulado en el software GNs3. Para IPv6 se utiliza el método de convivencia de túnel IPv6-IPv4. Las pruebas se llevaron a cabo durante 25 minutos.

Ilustración 115. Jitter MPLS IPv4-IPv6 emulado.

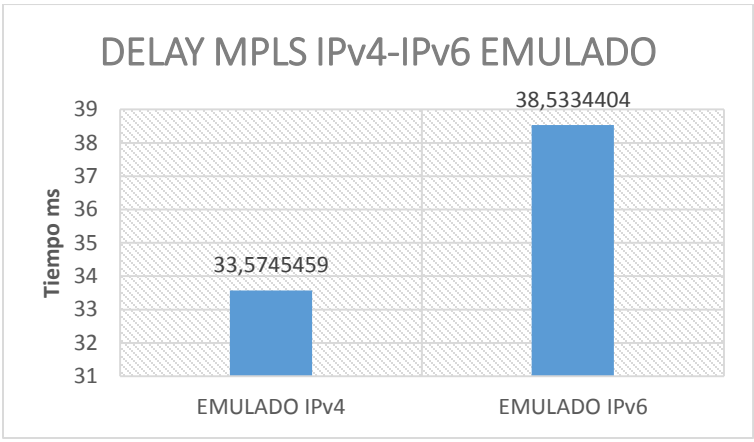


Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

En la figura anterior se puede evidenciar que el jitter en el sistema MPLS IPv6 es mayor que en el sistema MPLS IPv4, en mostrando un comportamiento similar a lo obtenido en los experimentos con equipos físicos. En los dos escenarios se presenta intermitencia con la diferencia que en MPLS IPv6 la llamada tiene un promedio de 18 a 20 minutos antes que se corte.

Para las pruebas realizas, los resultados de Delay que se obtuvieron en los sistemas MPLS IPv4 y MPLS IPv6 de forma emulada son los siguientes:

Ilustración 116. Delay MPLS IPv4-IPv6 emulado.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

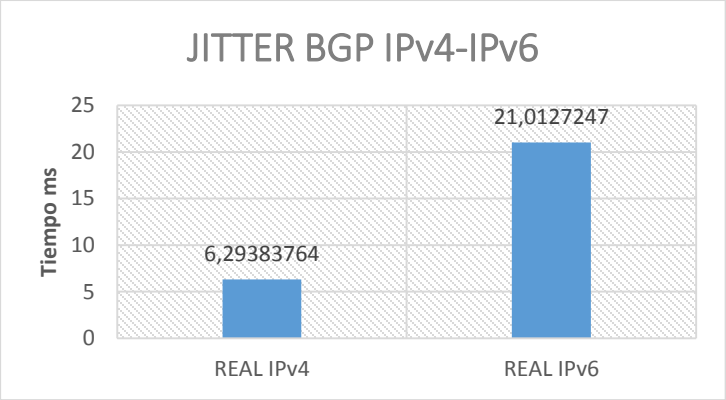
Se observa que en la variable Delay se obtiene el mismo comportamiento que para el Jitter donde el valor promedio del Dalay en el sistema MPLS IPv6 es mayor que en el sistema MPLS IPv4. De igual manera se aclara que si el método de

convivencia hubiera sido diferente como dual stack los resultados que se obtendrían serian diferentes a los de este proyecto.

6.7 ANÁLISIS JITTER Y DELAY BGP IPV4 Vs. IPv6 REAL.

Se observa a continuación el análisis comparativo entre el sistema BGP IPv4 y el sistema BGP IPv6, implementado en los routers Huawei. Para IPv6 se utiliza el método de convivencia de túnel IPv6-IPv4. Las pruebas se llevaron a cabo durante 25 minutos.

Ilustración 117. Jitter BGP IPv4-IPv6 real.

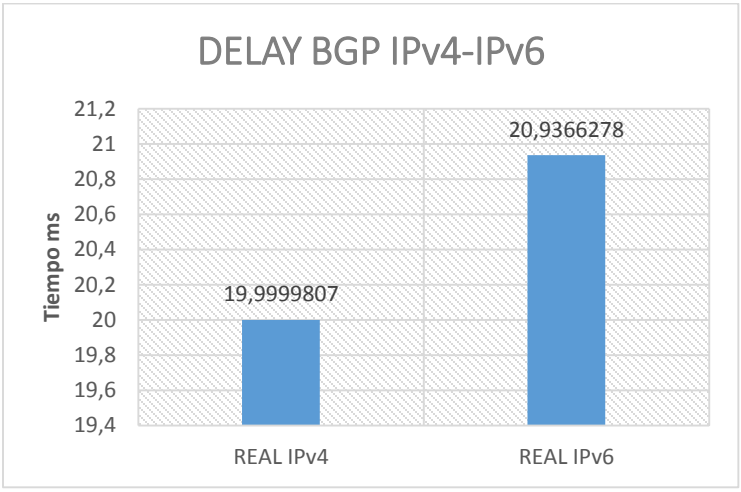


Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Al comparar el escenario basado en BGP IPv4 e IPv6. Se obtiene un comportamiento muy parecido al obtenido en el análisis de MPLS IPv4 e IPv6, por tanto se refuerza la observación que afirma que el método de convivencia túnel IPv4-IPv6 tiene efectos negativos en el Jitter.

Para las pruebas realizas, los resultados de Delay que se obtuvieron en los sistemas BGP IPv4 y BGP IPv6 de forma real son los siguientes:

Ilustración 118. Delay BGP IPv4-IPv6 real.



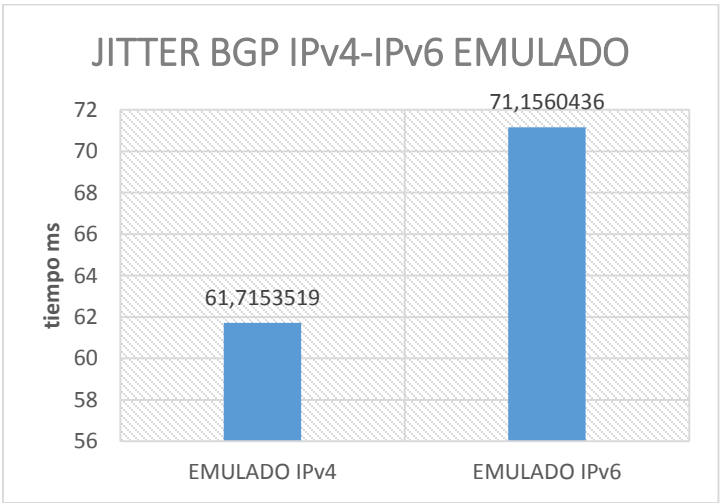
Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

En esta prueba tiene un comportamiento muy parecido al de MPLS IPv4 e IPv6, lo que confirma la observación que el túnel IPv4-IPv6 tiene un efecto negativo en el Jitter pero no muy profundo en el delay.

6.8 ANÁLISIS JITTER Y DELAY BGP IPV4 Vs. IPv6 EMULADO.

Se observa a continuación el análisis comparativo entre el sistema BGP IPv4 y el sistema BGP IPv6, implementado de manera emulado en el software GNs3. Para IPv6 se utiliza el método de convivencia de túnel IPv6-IPv4. Las pruebas se llevaron a cabo durante 25 minutos.

Ilustración 119. Jitter BGP IPv4-IPv6 emulado.

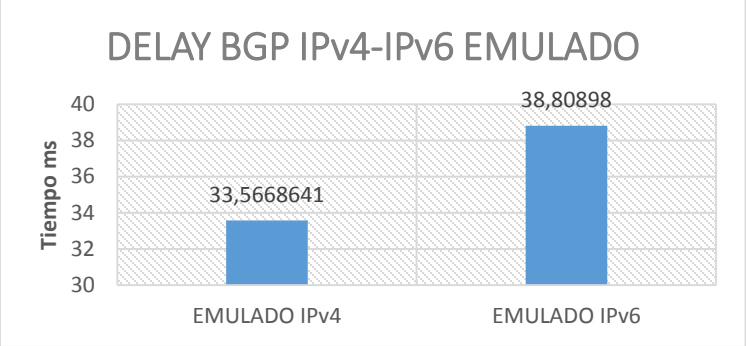


Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

En la figura anterior se puede evidenciar que el valor del jitter en el sistema BGP IPv6 es mayor que en el sistema BGP IPv4, pues al igual que las pruebas realizadas con router físicos se observa que el método de convivencia del túnel IPv6-IPv4 afecta directamente el jitter y la calidad de voz que se envía por el sistema. En los dos escenarios se presenta intermitencia con la diferencia que en BGP IPv6 la llamada tiene un promedio de 11 a 13 minutos antes que se cortar. El problema para IPV6 se presenta debido al túnel implementado y para ambos escenarios el equipo que se utiliza para el desarrollo de la prueba emulada.

Para las pruebas realizas, los resultados de Delay que se obtuvieron en los sistemas BGP IPv4 y BGP IPv6 de forma emulada son los siguientes:

Ilustración 120. Delay BGP IPv4-IPv6 emulado.



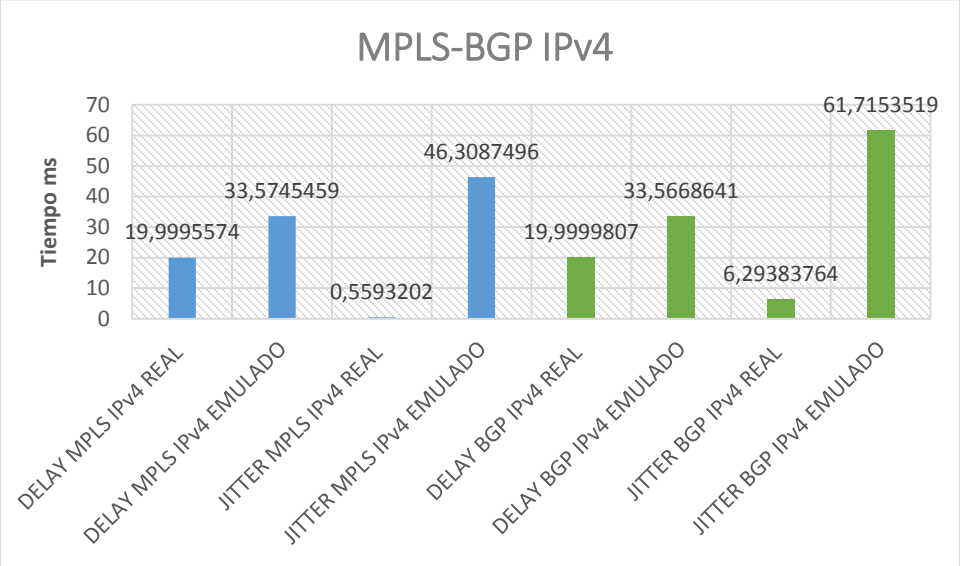
Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014.

Se observa que en la variable Delay se obtiene el mismo resultado que para el Jitter donde el tiempo en el sistema BGP IPv6 es mucho mayor que en el sistema BGP IPv4. Esto se presenta debía al método de convivencia implementado que es el túnel IPv6-IPv4 ya que baja la calidad de la voz en los sistemas.

6.5 ANÁLISIS JITTER Y DELAY MPLS Vs. BGP IPV4.

Se observa a continuación el análisis comparativo entre el sistema MPLS IPv4 y el sistema BGP IPv4.

Ilustración 121. MPLS Vs. BGP IPv4.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014

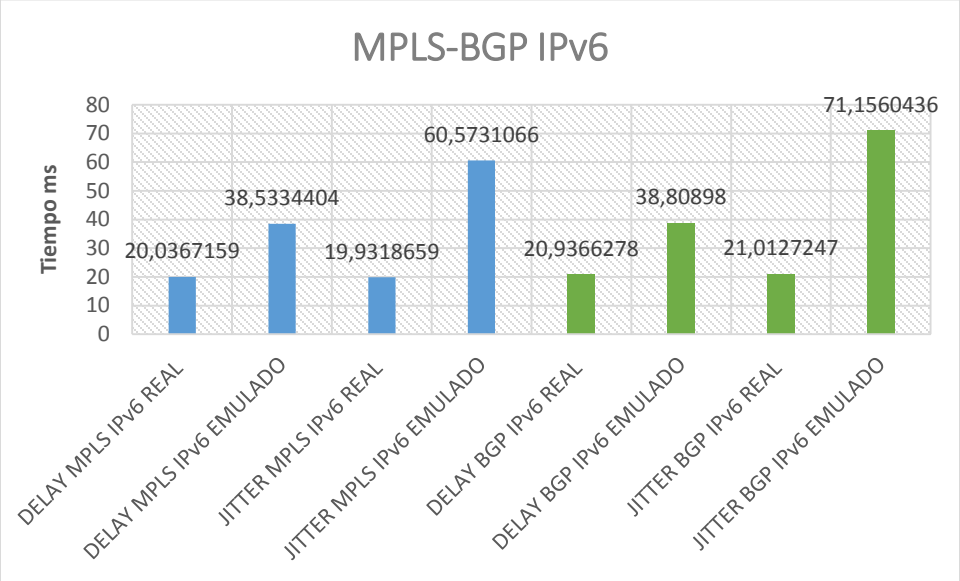
Se evidencia en la figura anterior que los sistemas implementados en los routers Huawei tienen mejor comportamiento que los sistemas emulados, debido a que los routers cuentan con un alto procesamiento, por el contrario que el equipo utilizado para emular los escenarios. También se identifica que el sistema MPLS respecto al jitter tiene más eficiencia al momento de brindar un servicio de tiempo real que el que brinda el sistema BGP.

6.5 ANÁLISIS JITTER Y DELAY MPLS Vs. BGP IPV6.

Se observa a continuación el análisis comparativo entre el sistema MPLS IPv6 y el sistema BGP IPv6.

Se evidencia en la siguiente figura que los sistemas implementados en los routers Huawei tienen mejor comportamiento que los sistemas emulados. Se observa que la diferencia de tiempo entre los sistemas no es muy grande, presentándose por la implementación del túnel como método de convivencia, la diferencia que hay es por el sistema que se está utilizando como base para el túnel, demostrando que MPLS tiene mejor rendimiento que BGP. Sin embargo no es el mejor escenario para brindar un servicio de voz IPv6 por los tiempos tan altos de jitter y delay afectando la calidad de la voz y en el caso emulado cortando la llamada.

Ilustración 122. MPLS Vs. BGP IPv6.



Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014

6.5 CUADRO COMPARATIVO DE RESULTADOS.

Tabla 13. Cuadro comparativo de los resultados.

	MPLS				BGP			
	IPv4 Real	**IPv4 Emulado**	**IPv6 Real (tunel)**	**IPV6 Emulado**	**IPv4 Real**	**IPv4 Emulado**	**IPv6 Real (tunel)**	**IPV6 Emulado**
Jitter (ms)	0,559	46,308	19,931	60,573	6,293	61,715	21,012	71,156
Delay (ms)	19,999	33,566	20,036	38,533	19,999	33,574	20,936	21,012
Comentarios	Como se puede observar en los resultados anteriores el rendimiento respecto al jitter del servicio de voz IP es mucho mejor en el sistema MPLS implementado sobre los routers físicos, ya que estos cuentan con mejor procesamiento que el software GNS3.		Como se puede observar en los resultados anteriores el rendimiento respecto a jitter del servicio de voz IP es mucho mejor en el sistema MPLS IPv6 implementado sobre los routers físicos, ya que estos cuentan con mejor procesamiento que el software GNS3. Aunque el jitter tiene un valor elevado debido al método de convivencia utilizado (tunel).		Como se puede observar en los resultados anteriores el rendimiento respecto al jitter del servicio de voz IP es mucho mejor en el sistema BGP implementado sobre los routers físicos, ya que estos cuentan con mejor procesamiento que el software GNS3. .		Como se puede observar en los resultados anteriores el rendimiento respecto a jitter del servicio de voz IP es mucho mejor en el sistema MPLS IPv6 implementado sobre los routers físicos, ya que estos cuentan con mejor procesamiento que el software GNS3. Aunque el jitter tiene un valor elevado debido al método de convivencia utilizado (tunel).	
	Respecto a los cuatro experimentos realizados sobre el sistema MPLS se evidencia que se obtiene un mejor rendimiento en la versión IPv4 real, comprobando que el método de convivencia para brindar el servicio en IPv6 no es la mejor solución de migrar tanto las redes como el servicio. En los experimentos emulados los tiempos son demasiados elevados por lo que son la opción más óptima para implementar un servicio en tiempo real debido al procesamiento necesario.				Respecto a los cuatro experimentos realizados sobre el sistema BGP se evidencia que se obtiene un mejor rendimiento en la versión IPv4 real, comprobando que el método de convivencia para brindar el servicio en IPv6 no es la mejor solución de migrar tanto las redes como el servicio. En los experimentos emulados los tiempos son demasiados elevados por lo que son la opción más óptima para implementar un servicio en tiempo real debido al procesamiento necesario.			

Fuente: Juan Sebastián Ramírez Moscoso. Facultad de Ingeniería de Telecomunicaciones, Universidad Santo Tomás, Bogotá, 2014

7. RECOMENDACIONES

Después del desarrollo del proyecto se obtuvieron conceptos relevantes para dar posibles soluciones en redes IPv4 e IPv6 que brinden servicios de tiempo real como la voz IP. Durante la implementación de los sistemas que aplican diferentes funcionalidades de los protocolos MPLS y BGP.

Teniendo en cuenta lo anterior se realizan una serie de recomendaciones para la implementación del servicio de voz IPv6 sobre sistemas MPLS y BGP:

- Debido al agotamiento en la actualidad del direccionamiento IPv4 los dispositivos están migrando a IPv6, por lo que se recomienda que al implementar un servicio de voz IPv6 se utilicen dispositivos de red que soporten IPv6 nativa (Dual Stack) ya que el método de convivencia del túnel de IPv6 sobre IPv4 reducen notablemente la calidad de la voz ya que el jitter está afectado directamente, como se pudo observar en el desarrollo de este proyecto con la implementación del túnel IPv6-IPv4.

- Si se pretende realizar un diseño de un sistema mediante emulaciones para brindar servicios de voz antes de implementarlo de forma real, se recomienda utilizar un computador con suficiente capacidad de procesamiento para obtener resultados relevantes que ayuden a prestar un servicio de buena calidad, ya que si las emulaciones se llevan a cabo con un computar con capacidad de procesamiento normal se obtendrán resultados como los evidenciados en este proyecto.

- Implementar sistemas MPLS para servicio de voz tiene gran beneficios ya que tiene un procesamiento más rápido de los paquetes, sin embargo los sistemas BGP es una buena opción, aunque reduce la calidad respecto al jitter, pero no es tan evidente si se usan dispositivos con un buen procesamiento.

- Tener en cuenta las variables Jitter y delay al momento de implementar una red para brindar servicios de voz IP, ya que estos lo afectan directamente, es decir si los tiempos son altos, el servicio tendrá mala calidad, escuchando la voz entre cortada con pedida de información y hasta la interrupción de la llamada.

8. FUNDAMENTACIÓN HUMANÍSTICA

Dentro de la filosofía de educación que impartía Santo Tomás de Aquino señala que la educación es el método de transformación del individuo de un estado imperfecto, incompleto y provisional a un estado más perfecto, valioso y superior al cual denomino virtud, resaltando que las virtudes de los seres humanos son hábitos repetitivos y de perfección de las facultades, proyectando al individuo a lo que es conveniente y bueno para la vida. (Escamilla., 2013.)

Basado en la misión de la Universidad Santo Tomás y el perfil profesional del ingeniero de telecomunicaciones, que consiste en un tomasino con formación ético e integral en el campo de la educación superior, en este caso en ingeniería de telecomunicaciones, brindando soluciones a problemáticas de la sociedad y el país de forma innovadora mediante acciones y procesos de enseñanza-aprendizaje, investigación y proyección social, respondiendo de manera ética, creativa y critica a las exigencias de la vida humana. (Tomás, 2013)

Teniendo en cuenta lo anterior el desarrollo de este proyecto se lleva acabo para brindar una solución viable a la sociedad en el entorno de la comunicación vía IP, ya que esta tecnología se implementa en la actualidad en IPv4 teniendo una gran demanda, el agotamiento del direccionamiento IPv4 produce una migración al direccionamiento IPv6 por lo que también se busca recomendar una forma viable para realizar esta migración, teniendo como base uno de los principios de Santo Tomás de Aquino, desarrollando conceptos que ayuden a un mejor bienestar de la sociedad.

CONCLUSIONES

En el desarrollo de este proyecto se utilizaron las metodologías y procedimientos requeridos para alcanzar los objetivos propuestos al iniciar este documento, logrando a su vez responder la pregunta problema que se estableció para análisis del rendimiento de voz sobre IPv4 e IPv6, realizando mediciones del jitter y delay en sistemas MPLS y BGP. De igual manera un tema relevante que se trató durante el desarrollo del documento fue el método de convivencia del túnel de IPv6 sobre IPv4, observando el rendimiento de la voz sobre este escenario.

De acuerdo a lo anterior se tienen las siguientes conclusiones:

- Al analizar el rendimiento de la voz sobre IPv4 e IPv6 utilizando como método de convivencia un túnel manual IPv6-IPv4 y realizando mediciones del jitter y delay en sistemas MPLS y BGP, se obtuvo que en el sistema MPLS IPv4 el jitter es menor en un 91,132084% que en el sistemas BGP IPv4, comprobando los resultados esperados teóricamente, ya que la estructura MPLS tienen un enrutamiento por labels evitando subir los paquetes hasta la capa 3.

- El método de convivencia del túnel no es la mejor opción para brindar un servicio de voz IPv6, ya que como se pudo observar en los resultados obtenidos el jitter en MPLS IPv4 es mucho menor en un 97,1938391% que en MPLS IPv6,se presente un caso similar para el sistema BGP IPv4 donde el jitter es menor en un 68,5508209 % que en BGP IPv6, recomendando que si se va implementar un servicio de voz IPv6 utilizar otro método de convivencia diferente al túnel para no presentar inconvenientes en la calidad de la voz.

- En la pruebas realizadas en IPv6 sobre los router Huawei se pudo evidenciar que el Jitter en MPLS era menor que en BGP, destacando que para los dos sistema se utilizó un túnel IPv6-IPv4 como método de convivencia reduciendo la calidad del servicio, ya que los routers con los que cuenta la Universidad no soportan VPN IPv6 nativa, como se implementó para IPv4.

- En los experimentos emulados se observó que los tiempos de Jitter y Delay eran muy altos para todos los escenarios afectando el servicio de voz que se estaba brindando, este efecto se presentó ya que el software GNs3 no cuenta con la misma capacidad de respuesta que los router Huawei, por lo que GNs3

se manejaría más como una herramienta para realizar pruebas de configuración y para mostrar tendencias de los sistemas implementados. Sin embargo se obtiene un comportamiento similar a los obtenidos en los equipos reales

- Se observó que la variable Delay en los sistemas MPLS y BGP tiene valores muy similares sin embargo MPLS sigue teniendo un mejor rendimiento, este se presenta ya que en los experimentos diseñados para este proyecto el servicio se afecta debido a la variable Jitter.

- Se pudo observar durante el desarrollo del proyecto que el protocolo se señalización SIP funciona sin inconvenientes sobre diferentes sistemas sin importar el protocolo IP que está establecido (IPv4 o IPv6), para los experimentos en IPv6 el registro se llevó a cabo por medio de un servidor DNS sin afectar el funcionamiento del servicio de voz.

- Respondiendo la pregunta problema establecida ¿cuál es el impacto con respecto a jitter y delay al implementar IPv4 e IPv6 en un servicio de voz en sistemas MPLS y BGP?, el impacto obtenido es que el método de convivencia del túnel es negativo para brindar un servicio de voz IPv6, bajando la calidad de la voz, como se pudo observar hasta la caídas de las llamadas en tiempos específicos en los escenarios emulados. A su vez se pudo observar que el sistema MPLS tiene un mejor rendimiento que BGP para brindar servicios de tiempo real, viéndose afectado por el tiempo de las variables jitter y delay.

V. BIBLIOGRAFIA

Asterisk. (014 Digium, Inc. All Rights Reserved.). *asterisk*. Obtenido de asterisk: http://www.asterisk.org/

Baroncelli, F. C., Martini, V., & Castoldi, P. (2007). A distributed signaling for the provisioning of on-demand VPN services in transport networks.

Chen Shengyang State Key Lab. of Networking & Switching, B. U. (2010). P2P Communication Mechanism Based on Request Forwarding in IPv4 and IPv6 Coexistence Network.

Daisaku Shimazaki, T. M. (october 2008). Advanced Operation with MPLS traffic engineering for IP over GMPLS-based Optical Network.

Dirceu Cavendish, H. O. (october 2004). Operation, Administration, and Maintenance in MPLS Networks.

Dr. Reyadh Shaker Nahum, M. M. (2013). Performace Evaluation for VoIP and MPLS. *IEEE*.

Escamilla., H. J. (2013.). Tomás de Aquino y el tratado de la educación. Bogotá.

fit, D. D. (1998). *TCP/IP: Arquitectura, protocolos e implementación con IPv6 y seguridad de IP.* Estados Unidos.

Ge Zhang, S. F.-H. (2009). *Revealing the Calling History of SIP VoIP Systems by Timing Attacks.* Karlstad University, Fraunhofer FOKUS, Karlstad, Sweden - Berlin, Germany.

Gil Cabezas, J. (2009). Protocolo de Transporte en Tiempo Real .

GNs3. (opyright © 2007-2014 GNS3. All Rights Reserved). *http://www.gns3.com*. Obtenido de http://www.gns3.com: http://www.gns3.com

GONZÁLEZ, W. S. (2011). PROTOCOLOS DE SEÑALIZACIÓN USADA ACTUALMENTE PARA TERMINALES MÓVILES E IP. *GRUPO DE INVESTIGACIÓN PROMENTE LÍNEA DE INVESTIGACIÓN EN TELECOMUNICACIONES*. bogotá: FUNDACIÓN UNIVERSITARIA KONRAD LORENZ.

Haitao Zheng, B. J. (2002). An improved UDP protocol for video transmission over Internet-to-wireless networks. *3*.

Hongqiang Li, Q. S. (27-29 June 2011). Research and application of a UDP-based reliable data transfer protocol in wireless data transmission.

ITU, T. S. (2000). SERIES G: TRANSMISSION SYSTEMS AND MEDIA, DIGITAL SYSTEMS AND NETWORKS .

Lambrinos, L. D. (4-9 March 2007). Integrating Voice over IP Services in IPv4 and IPv6 Networks.

Lu Tian1, N. D. (October 2011). Study of SIP Protocol Through VoIP Solution of “Asterisk”.

Martinez, R. F. (24 noviembre del 2006). *procolo multiple por conmutacion de etiquetas.* cuenca: Universidad Politecnica Salesiana.

Mazin Alshamrani, H. C. (April 2013). Evaluation of SIP Signalling and QoS for VoIP Over OLSR MANET Routing Protocol.

Mejía, Ó. Á. (2011). *Migración del protocolo IPv4 a IPv6.* Mexico: Departamento de Ingeniería eléctrica UAM.

Ming Luo, Y.-Y. W. (2008). A Certificate-Based Authenticated Key Agreement Protocol for SIP-Based VoIP Networks.

Narayan, S. D., Branks, C., & Fan, L. (25-27 June 2010). Network performance evaluation of VoIP on windows desktop operating systems with IPv4 and IPv6 network implementations. *Computer Design and Applications (ICCDA), 2010 International Conference on (Volume:5)*.

Octavio Salcedo, D. L. (2012). From voice over IPv4 to voice over IPv6 networking migration. *Revista Tecnura*, 76-87.

Óscar, Á. M. (2011). *migración del protocolo IPv4 a IPv6.* Ciudad de México D.F: Universidad Autónoma de Mexico.

Radha, B., Dept. of Comput. Sci. & Eng., N. I., & Selvakumar, S. (2011). DEEPAV2: A DNS monitor tool for prevention of public IP DNS rebinding attack.

Rekhter, B. D. (2000). MPLS Technology and Applications.

Simon ZNATY, J.-L. D. (2005). SIP : Session Initiation Protocol.

Singh, I., Lamabam, P., & Devi, N. (21-22 Feb. 2014). A technique for classification of VoIP flows in UDP media streams using VoIP signalling traffic.

Taemoor Abbasi, S. P. (2005). A COMPARATIVE STUDY OF THE SIP AND IAX VOIP PROTOCOLS.

Tomás, U. S. (2013). Facultad de Ingeniería de Telecomunicaciones. Reglamento Estudiantil y Proyecto Educativo. Bogotá.

Tran Cong Hung Posts & Telecommun. Technol. Inst., H. C., Thang, N. D., & Huy, T. D. (12-14 Feb. 2007). Interoperability between Mobile IPv4 and Mobile IPv6 based on MPLS core network. *Advanced Communication Technology, The 9th International Conference on (Volume:2)*.