ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO

FACULTAD DE INFORMÁTICA Y ELECTRÓNICA

ESCUELA DE INGENIERÍA ELECTRÓNICA Y COMPUTACIÓN

ESTUDIO COMPARATIVO DE VoIP Y TELEFONÍA IP EN IPV6 E IPV4

CASO PRÁCTICO: IMPLEMENTACIÓN DE CENTRAL DE TELEFONIA IP.

"TESIS DE GRADO, PREVIA OBTENCIÓN DEL TÍTULO DE INGENIERO EN ELECTRÓNICA Y COMPUTACIÓN"

DESARROLLADO POR:

DANNY ROBERTO CÁCERES MEZA

OSVALDO PAUL ORTIZ MALDONADO

RIOBAMBA – ECUADOR

2010

Agradezco a mis padres quienes me formaron como la persona que soy hoy en día, forjaron en mí el carácter y losvalores necesarios para salir adelante yquienes me han dejado este gran legado que durará por siempre conmigo. A mi esposa Carolina quien estuvo siempre alentándome y apoyándome incondicionalmente.

Danny Cáceres

Le agradezco a Dios por todas las bendiciones recibidas en la realización de esta gran meta de mi vida. A mis Padres por haberme inculcado que las grandes conquistas son el resultado de muchos sacrificios, dedicación y entrega permanentes y porque he sentido su presencia cariñosa en todo momento, a mi hermano Jorge por estar pendiente de todas mis cosas.

Osvaldo Ortiz M.

Expresamos un profundo agradecimiento a las Autoridades y Maestros de la Escuela de Ingeniería Electrónica de manera especial a los Ingenieros Daniel Haro y Marcelo Donoso, porque con su valiosa guía y el aporte de sus conocimientos y experiencias hicieron posible la realización de este Trabajo Investigativo, con sentimientos de singular orgullo por pertenecer a una Institución en donde la Calidad y la Excelencia son su carta de presentación, La Escuela Superior Politécnica del Chimborazo.

Danny Cáceres

Osvaldo Ortiz M.

Dedico este Trabajo Investigativo a mi hijo Dániel Cáceres ese pequeño ángel que vino a cambiar mi vida ypor quien luché hasta el final por terminar mi carrera y por quien sigoluchando para culminar mis objetivos.

Danny Cáceres

Dedico este Trabajo a mi Papi Jorge, que desde el cielo se alegrará al culminar este sueño que tanto anheló compartir conmigo mientras vivía.

A María mi Mami querida por ser mi amiga y Compañera incondicional, a mi hermano Jorge y a Andrea mi futura esposa.

Osvaldo Ortiz M.

"Nosotros, Danny Roberto Cáceres Meza y Osvaldo Paúl Ortiz Maldonado, somos los responsable de las ideas, doctrinas y resultados expuestos en esta tesis y el patrimonio intelectual de la misma pertenecen a la Escuela Superior Politécnica de Chimborazo".

INDICE DE ABREVIATURAS

CODEC	Fusión de "Compresor – Descompresor" que describe un dispositivo o programa que convierte un conjunto de datos o una señal.
E1 / T1	Estándar de telecomunicaciones para transmisión simultánea de voz (E1 – 32 canales, T1 – 24Canales)
GPL	General Public License(Licencia Pública General)
ICR	Intelligent Call Routing Routing (Enrutamiento Inteligente de Llamadas)
IP	Internet Protocol (Protocolo de Internet)
IP-PBX	Central Privada que utilice Protocolo de Internet.
ISP	Internet Service Provider (Proveedor de Servicio de Internet)
ITSP	Internet Telephone Service Provider (Proveedor de Servicio Telefónico por Internet)
LAN	Local Area Network (Red de Area Local)
MOS	Mean OpinionScore (Opinión de la Puntuación Medida)
PBX	Private Branch Exchange (Central Privada)
PSTN	Public Switching Telephone Network (Red Pública de Telefonía

	mundial)°
RTP	Real-time Transport Protocol (Protocolo de Transporte en Tiempo Real)
SIP	Session Initiation Protocol (Protocolo de Inicio de Sesiones)
TCP	Transmission Control Protocol (Protocolo de Control de Transmisión)
UDP	User Datagram Protocol (Protocolo de Datagrama a Nivel Usuario)
VoIP	Voice Over Internet Protocol (Voz sobre Protocolo de Internet)
QoS	Quality of Service (Calidad de Servicio)
IANA	Internet Assigned Numbers Authority (Autoridad de Asignación de Números de Internet)
NAT	Network Address Translation (Traducción de Dirección de Red)
CIDR	Classless Inter-Domain Routing (Encaminamiento Inter-Dominios sin Clases)
NDP	Neighbor Discovery Protocol (Protocolo de descubrimiento de vecinos)

INDICE GENERAL

INDICE DE FIGURAS

INDICE DE TABLAS

RESUMEN

Se ha comparado el rendimiento de VoIP en una red LAN IPv6 e IPv4 con la finalidad de observar el funcionamiento yconfiguración de IPv6 y rescatar los beneficios en relación a IPv4, para lo cual se realizaron pruebas de rendimiento y funcionamiento de VoIPen cada protocolo utilizando para esto el sniffer Wireshark, el análisis de la implementación y la evaluación del sistema mediante encuesta.

Las medidas de rendimiento analizadas fueron el jitter, el retardo,la pérdida de paquetes, conlos datos se calculó el valor MOS de cada protocolo; escalabilidad, facilidad de implementación y la validación del sistema por parte de una muestra de usuarios.

Se determinóque los valores medios de retardo para IPv4 e IPv6 son similares, el jitter para IPv4 es ligeramente superior a la de IPv6, estos factores y el códec se utilizó para el cálculo de MOS, el índice nos dio como resultado una pequeña superioridad de IPV6.A niveles moderados de tráfico de fondo, la relación entre el rendimiento IPv4/IPv6 fue cercano al ideal (teórica).

Los resultados obtenidos de las evaluaciones del sistema junto con los demás factores muestran una superioridad de la implementación de un sistema de VoIP sobre IPv6.

La comunicación de voz a través de la red IP, de manera económica y efectiva, es un hecho, por lo que se recomienda que las empresas utilicen las posibilidades que les ofrece el protocolo IPv6 para incrementar la productividad y competitividad.

SUMMARY

The yield of VoIP has been compared in a net LAN IPv6 with the purpose of the observe the operation and configuration of IPv6 and to rescue the benefits in relation to IPv4, for this reason they were carried out yield rests and operation of VoIP in each protocol using for this, the sniffer Wireshark, the analysis of the implementation and evaluation of the system using a survey.

The analyzed yield measures were the jitter, the retard, the lost of packages, with the data it was calculated the value MOS of each protocol; scalability, implementation, easiness and the validation of the system on the part of a sample of users.

It´s determinated that the values retard means for IPv4 and IPv6 are similar, the jitter for IPv4 is lightly superior to that IPv6, these factors and the codec that we use for the calculate of MOS, this index gave us a result a small superiority of IPv6. At moderate levels of a traffic of bottom, the relationship among the yield IPv4/IPv6 reached near to the ideal (Theoretical).

The obtained results of the evaluations of the system together with the other factors show a superiority of the implementation of a system of VoIP over IPv6 it has more than enough.

The voice communication through the net IP, in an economic and effective way, for these reasons we recommended that the companies use the possibilities that this protocol IPv6 offers them to increase the productivity and competitivities.

INTRODUCCIÓN

Hace 30 años Internet no existía, y las comunicaciones se realizaban por medio del teléfono a través de la red telefónica pública conmutada (PSTN), pero con el pasar de los años y el avance tecnológico han sido posible implementar nuevas tecnologías de comunicación.

Hoy por hoy podemos ver una gran revolución en comunicaciones, todas las personas usan los computadores e Internet en el trabajo y en el tiempo libre para comunicarse con otras personas, para intercambiar datos y a veces para hablar con más personas usando aplicaciones como NetMeeting o Teléfono IP, el cual particularmente comenzó a difundir en el mundo la utilización de la comunicación en tiempo real por medio del PC: VoIP Con el avance de la tecnología al día de hoy alcanzamos un servicio VoIP de altas prestaciones y calidad en tiempo real. No obstante, si en una empresa se dispone de una red de datos, también se podría pensar en la utilización de esta red para el tráfico de voz entre las distintas delegaciones de la empresa.

Las ventajas que se obtendrían al utilizar la red para transmitir tanto la voz como los datos son evidentes, ahorro de costos de comunicaciones, pues las llamadas entre las distintas delegaciones de la empresa saldrían gratis.

En sus inicios una persona conectaba manualmente cables para establecer comunicaciones en lo que era conocido como un PMBX (PBX Manual). Este dispositivo fue reemplazado por uno dispositivo electromecánico automático y sistemas electrónicos de conmutación llamados PABX (PBX automático) que desplazaron al PMBX hasta hacerlo casi inexistente, entonces los términos PABX y PBX se convirtieron en sinónimos.

Asterisk es una aplicación de código abierto, una central telefónica (PBX). Como cualquier PBX, se puede conectar un número determinado de teléfonos para hacer llamadas entre sí e incluso conectar a un proveedor de servicio o bien a una RDSI.

Asterisk tiene licencia GPL originalmente desarrollado para el sistema operativo Linux pero actualmente también funciona en BSD, MacOSX, Solaris y Microsoft Windows aunque la plataforma nativa (Linux) es la mejor soportada de todos.

Asterisk incluye muchas características anteriormente solo disponibles en costosos sistemas propietarios PBX.

Una de las características importantes de Asterisk es que soporta muchos protocolos VoIP como pueden ser SIP, H.323, IAX y además posee soporte experimental para el protocolo IPv6.

Asterisk puede interoperar con teléfonos IP actuando como un registrador y como Gateway entre ambos.

Internet, gracias al auge de la pila de protocolos TCP/IP, ha traído grandes avances y muchas posibilidades de servicios y aplicaciones que pueden usar esta red. Sin embargo, se presentan varios problemas como es el funcionamiento en modo best-effort, lo que no permite dar calidad de servicio a las aplicaciones de tiempo real, como VoIP, la falta de direcciones IPv4 clase B, demasiados sistemas conectados, demasiadas entradas en las tablas de routing, incremento progresivo en el tiempo de búsqueda, DNS, etc, situación salvada temporalmente con NAT.

La siguiente generación del protocolo Internet, la versión IPv6 ofrece muchas ventajas como son la interoperación con la versión actual IPv4, expansión en las capacidades de routing y direccionamiento, autoconfiguración, mayor

seguridad, etc. Además, ofrece una plataforma para la nueva funcionalidad de Internet que será necesaria en un futuro inmediato.

La adopción de IPv6 ha sido un proceso lento. A la fecha, el tráfico IPv6 en Internet representa menos de un 1% del total cursado. Aun cuando diversos estudios (Loshin) pronostican que en pocos años más se producirá el agotamiento total de las direcciones IPv4.

La necesidad de migrar a IPv6 está originada por las nuevas tendencias en el mundo actual de las telecomunicaciones, debido al incremento de usuarios, desarrollo de avanzados sistemas y la convergencia de voz, vídeo y datos, en infraestructuras basadas en IP.

El propósito de esta tesis es presentar el análisis comparativo del funcionamiento respectivo de un sistema de Telefonía IP bajo el protocolo IPv4 e Ipv6 ya que se demostró los beneficios que nos ofrece el nuevo protocolo, siendo este documento un referente para la inminente migración de la Telefonía IP a la utilización del protocolo IPv6.

DESCRIPCIÓN DEL TRABAJO

Para cumplir con los objetivos del presente trabajo de investigación, se lo ha dividido en cuatro capítulos:

El Primer Capítulo: Marco Referencial, explica los fundamentos de la tesis y se describen los objetivos.

El Segundo Capítulo: Marco Teórico, presenta los diversos conceptos necesarios para el correcto entendimiento de la tesis.

El Tercer Capítulo: Marco Metodológico, analiza mediante implementaciones aisladas las tecnologías descritas en el capítulo anterior.

El Cuarto Capítulo: Análisis e Interpretación de los Datos, muestra la implementación de los servidores, las pruebas de desempeño a las que fueron sometidos y sus resultados.

Finalmente se llega a describir las Conclusiones y Recomendaciones, se incluyen la síntesis de los resultados obtenidos, presentados en capítulos anteriores y sugerencias finales del trabajo de investigación.

CAPÍTULO 1 I

MARCO REFERENCIAL

1.1 PLANTEAMIENTO DEL PROBLEMA

Las redes VoIP han alcanzado un grado razonable de éxito comercial. Las empresas están utilizando la tecnología para ahorrar dinero en costos de trunking y para mejoras funcionales, tales como la movilidad, relacionadas con las funciones de presencia y mensajería unificado.

Sin embargo, algunos problemas fundamentales se oponen a la escalabilidad ilimitada de VoIP. El primer problema es la falta de calidad de servicio en muchas redes IP. El segundo problema se refiere a la integridad de extremo a extremo de la señalización de VoIP y las rutas de acceso. Es difícil de llevar paquetes de VoIP a través de firewalls, no sólo por consideraciones de protocolo, sino también por la traducción de direcciones de red (NAT). Los problemas de seguridad, como escuchas telefónicas y la piratería, son otro problema potencial. La siguiente generación de redes VoIP basadas en IPv6 se

encuentran ahora en la mesa de dibujo para abordar estas cuestiones, en concreto la escalabilidad y la fiabilidad.

Se desea implementar un prototipo de central de telefonía IP que trabaje bajo el protocolo IPv4 e IPv6, comparar, analizar y demostrar el mejor rendimiento de la telefonía IP en redes IPV6.

1.2 JUSTIFICACIÓN DE LA INVESTIGACION

En las eventuales congestiones producidas por el tráfico entre dos equipos (host o terminal) de distintas redes, cada paquete de información compite por un poco de ancho de banda disponible para poder alcanzar su destino.

Típicamente, las redes IPv4 operan en la base de entrega del mejor esfuerzo, donde todo el tráfico tiene igual prioridad de ser entregado a tiempo. Cuando ocurre la congestión, todo este tráfico tiene la misma probabilidad de ser descartado. En ciertos tipos de datos que circulan por las redes hoy en día, por ejemplo tráficos con requerimientos de tiempo real (voz o video), es deseable que no ocurra pérdida de información, que exista un gran ancho de banda disponible y que los retrasos en los envíos de estos paquetes de datos sean mínimos. Es por ello que surge la necesidad de aplicar Calidad de Servicio (QoS) en el nivel del transporte de datos, métodos de diferenciación de tráficos particulares con el fin de otorgar preferencia a estos datos sensibles, además IPv6 (Internet Protocol Version 6), contiene nuevas y reestructuradas especificaciones para ejercer QoS.

La versión IPv6 puede ser instalada como una actualización de software en los dispositivos de red de Internet e interoperar con la versión actual IPv4.

La necesidad de migrar a IPv6 está originada por las nuevas tendencias en el mundo actual de las telecomunicaciones y además en IPv6 el encaminamiento en la red troncal es más eficiente, debido a una jerarquía de direccionamiento basada en la agregación y a que la fragmentación y desfragmentación de los paquetes se realiza extremo a extremo.

Aunque el IPv6 aún no está operativo de forma oficial, cada día más aplicaciones lo soportan, abriéndose paso a la futura convergencia al nuevo sistema. Si bien aún más de 85% de los sistemas solo funcionan bajo IPv4. Asterisk es una de las aplicaciones que está trabajando para que sea compatible con IPv6.

Actualmente existe una distribución para darle soporte experimental a Asterisk para que funcione bajo IPv6.

Asterisk es el más popular y extensible sistema telefónico de código abierto en el mundo, ofreciendo flexibilidad, funcionalidad y características no disponibles en sistemas de propiedad empresarial avanzados, de alta gama.

Asterisk-IPv6 muestra el poder de VoIPv6 evitando todas las cuestiones relacionadas con NAT cuando se utiliza IPv4. La presencia de NAT para VoIPv4 provoca problemas en los usuarios, tales como la no conexión de llamadas, audio en un solo sentido, no trabaja DTMF. Asterisk-IPv6 resuelve todas estas cuestiones y también aporta, junto con IPv6, la movilidad real de IP, seguridad y autoconfiguración.

La tecnología de voz sobre IP (VoIP) rápidamente está reemplazando a la telefonía convencional en todo el mundo. Esto se debe, principalmente a sus bajísimos costos y a la cantidad de funciones disponibles.

En nuestro País existen cada día más empresas e instituciones que requieren un sistema de comunicación interno rápido, fiable, de bajo costo, acorde a la tecnología actual y porque no, con vistas al futuro; por este motivo es muy importante adquirir conocimientos para la implementación de sistemas bajo el nuevo protocolo IPV6, aprovechando sus ventajas y tomando en cuenta su inevitable introducción a todos los sistemas basados en su predecesor.

Este documento pretende ser una fuente de información para futuras investigaciones, además un punto de partida para la transición al protocolo IPV6, a todos los servicios y ventajas que puede ofrecer.

1.3 OBJETIVOS DE LA INVESTIGACION

1.3.1 Objetivo General

Estudiar comparativas y el funcionamiento de la Telefonía IP en IPv6 conjuntamente con IPv4.

1.3.2 Objetivos Específicos

- ✓ Investigar el funcionamiento y configuración de IPv6 y rescatar los beneficios en relación a IPv4.
- ✓ Implementar un prototipo de una central telefónica en IPv4 e IPv6.
- ✓ Investigar el funcionamiento y configuración del Software IP PBX Asterisk.

- ✓ Determinar el softphone que cuente con todas las características necesarias para la implementación del sistema de telefonía IP en cada protocolo.
- ✓ Evaluar los sistemas implementados mediante el uso de una encuesta.
- ✓ Determinar los factores de rendimiento y calidad de cada sistema mediante el uso de un sniffer.

1.4 ALCANCE

En el presente trabajo se estudiará las redes IPv4 e Ipv6, instalación, configuración y funcionamiento de los IP PBX, se prestara mayor atención a la velocidad de transmisión real y características de calidad presentados en los mismos.

Una vez entendido el funcionamiento de la IP PBX tanto en IPv4 como en Ipv6 se los someterá a un escenario que ponga a prueba sus características teóricas, esto con el objetivo de estudiar sus capacidades y limitaciones al trabajar bajo un mismo entorno.

Finalmente utilizando los parámetros establecidos en el escenario se tendrá que establecer las diferencias, beneficios y problemas entre el funcionamiento de una IP PBX bajo los protocolos de red IP.

1.5 RECURSOS

1.5.1 Recursos Humanos

Se contará con la colaboración de:

- ✓ Desarrolladores
- ✓ Tutor de Tesis
- ✓ Administradores de red de la UNACH
- ✓ Colaboradores

1.5.2 Recursos Técnicos

Hardware

Tabla 1-I: Recursos Hardware

Equipo	Características	Estado
Computador	Procesador: Core2Duo 2.2 GHz Memoria RAM: 1GB Disco Duro: 160GB	Bueno
Portátil	Procesador: Core2Duo 1.6 GHz Memoria RAM: 1GB Disco Duro: 120GB	Bueno
Portátil	Procesador: Core2Duo 1.6 GHz Memoria RAM: 1GB Disco Duro: 160GB	Optimo
Impresora	Laser Samsung ML-1740	Bueno

Impresora	Inkjet Lexmark X1270	Bueno
LAN Tester	-	-
Routers	-	-

Fuente: Los Autores de esta investigación

Software

Tabla 1-II: Recursos Software

Nombre	Descripción	Estado
Windows XP SP3	Sistema Operativo	No Legal
Centos 5.3	Sistema Operativo	Libre
Asteriskv6	Gestor Sistema Telefónico	Libre
Microsoft Word 2007	Procesador de Textos	No Legal
Microsoft Project 2007	Gestor de Proyectos	No Legal
Microsoft Visio 2007	Creación de Diagramas	No Legal
Wireshark 1.2.7	Capturador de Paquetes	Libre
Linphone 3.2.1	Softphone	Libre

Fuente: Los Autores de esta investigación

1.5.3 Otros

Tabla 1-III: Otros Recursos

Categoría	Material(es)
Informativo	Libros Revistas Sitios Web Documentos Digitales
Respaldo de Información	Hojas Cuadernos Copias Impresiones
Almacenamiento de Información	Pen Drives CDs DVDs
Escritura	Lápices

	Esferográficos

Fuente: Los Autores de esta investigación

CAPÍTULO 2 II

MARCO TEÓRICO

2.1 LA NECESIDAD DE IPV6

El protocolo de Internet (IP) es un protocolo no orientado a la conexión usado para trasmitir información a través de una red de paquetes conmutados. Se ubica en la capa 3 del modelo ISO/OSI y su función es entregar paquetes desde un nodo de origen a uno de destino, basado en la dirección escrita en cada paquete. El protocolo de Internet versión 4 (IPv4) es la cuarta iteración del protocolo IP y la primera versión en ser utilizada en ambientes de producción. Es el protocolo dominante en Internet, utilizado para conectar redes de forma interna y hacia el exterior. Dentro de sus principales características se encuentran:

- ✓ Enrutamiento y direccionamiento: Provee una dirección única a cada dispositivo de una red de paquetes. IPv4 fue especialmente diseñado para facilitar el enrutamiento de información (paquetes) a través de redes de diversa complejidad.

- ✓ Encapsulación: El protocolo IPv4 nace como una división del antiguo protocolo TCP ("Transmission Control Protocol"). Se ubica en la capa 3 del modelo ISO/OSI y puede funcionar sobre diversos protocolos de nivel inferior.
- ✓ Mejor esfuerzo: El protocolo IP provee un servicio de transmisión de paquetes no fiable (o de mejor esfuerzo). No se asegura que los paquetes enviados lleguen correctamente al destino.

La versión de IPv4 usada actualmente en Internet no ha cambiado sustancialmente desde su publicación inicial en 1981. IPv4 ha demostrado ser un protocolo robusto, fácil de implementar y con la capacidad de operar sobre diversos protocolos de capa 2. Si bien fue diseñado inicialmente para interconectar unos pocos computadores en redes simples, ha sido capaz de soportar el explosivo crecimiento de internet. Sin embargo en el último tiempo, se han hecho notar diversos problemas existentes en IPv4, asociados al crecimiento de Internet y a la aparición de nuevas tecnologías y servicios que requieren conectividad IP.

2.1.1 Agotamiento direcciones IP

Una dirección IPv4 tiene un tamaño de 32 [bit], los que permiten un máximo teórico de 232 (4.294.967.296) direcciones a asignar. En los inicios de Internet, se utilizaron métodos de distribución poco eficientes, como la asignación por clases, mediante los cuales se asignaron grandes bloques de direcciones a organizaciones que solo requerían unas pocas. Esto ha generado que

actualmente muchas organizaciones posean un gran número de direcciones que no se encuentran utilizadas.

Los primeros reportes de alerta sobre el inminente agotamiento de direcciones IP se dieron a conocer alrededor de 1990. Diversas soluciones y protocolos han permitido extender la vida útil de IPv4, tales como la traducción de direcciones de red (NAT), el enrutamiento sin clases entre dominios (CIDR) y el uso de asignaciones temporales de direcciones con servicios tales como DHCP y RADIUS/PPP. Actualmente, se ha establecido una política jerarquizada para la asignación de direcciones IPv4, en donde el IANA ("Internet Assigned Numbers Authority") tiene a su cargo el manejo de los bloques de direcciones IPv4 que se encuentran libres. Junto al IANA, se encuentran los registros regionales de Internet (AFRINIC, APNIC, ARIN, LACNIC y RIPENCC) quienes reciben bloques de direcciones delegados por el IANA y los distribuyen entre los proveedores de servicios (ISP) de la región del mundo que administran.

El IANA asigna bloques de prefijo /8, (equivalentes a 1/256 del total de direcciones) a los registros regionales. Dado que el rango de direcciones comprendido entre 224.X.X.X y 239.X.X.X se encuentra reservado para tráfico "multicast", y el rango entre 240.X.X.X y 254.X.X.X se encuentra reservado para trabajos experimentales, el espacio real de direcciones disponibles para ser asignadas es de 223 bloques /8, los cuales representan 16.777.214 direcciones cada uno. En la Figura 2.1 se observa la distribución actual[1] de bloques /8.

1Datos al 20/11/2009

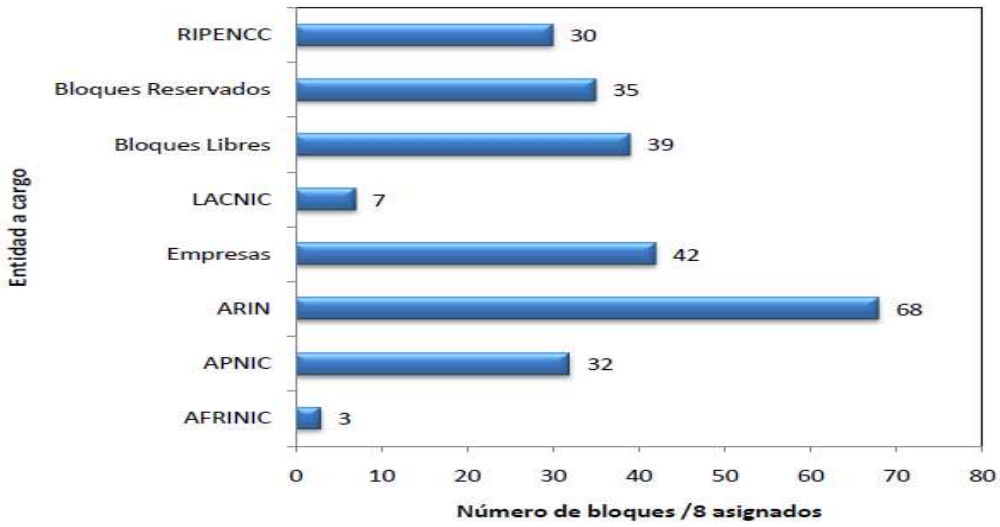


Figura 2-I: Distribución actual de Bloques /8

En la Figura 2.1 se observa que la mayor parte de los bloques se encuentra asignado al registro regional ARIN, que distribuye direcciones a Canadá, EE.UU. e Islas del Noratlántico. Se puede apreciar que una parte importante de los bloques /8 se encuentran asignados directamente a empresas y organizaciones, quienes recibieron dichos bloques como producto de las políticas de asignación anteriores a 1993. Dentro de los grupos reservados, se encuentran los bloques asignados a direcciones IP privadas, tráfico "multicast" y otros usos aun no definidos. Los 39 bloques libres son manejados directamente por el IANA, quien los delega a cada registro regional de acuerdo a sus requerimientos.

Es complicado estimar la fecha exacta en que se agotarán todas las direcciones IPv4 disponibles, ya que diversos factores pueden adelantar o retrasar dicha fecha. Dentro de esos factores se encuentran posibles cambios en la política de asignación, recuperación de bloques no

utilizados o incluso la venta de direcciones IP entre privados. Una de las fuentes más utilizadas para proyectar el agotamiento de direcciones IPv4 es el sitio "IPv4 Address Report", que a partir de la información publicada por el IANA y los registros regionales, entrega una fecha estimada de agotamiento de direcciones IPv4.

En la Figura 2.2 se presenta una proyección del agotamiento de bloques /8. Este análisis modela el comportamiento de cada registro regional, considerando su demanda histórica de bloques de direcciones IP. En la figura se observan tres curvas, una asociada a los bloques asignados a registros regionales ("Assigned"), otra que representa aquellos bloques asignados que son anunciados efectivamente hacia internet ("Advertised") y una que señala aquellos bloques asignados que no son anunciados ("Unadvertised").

En base a estas proyecciones, se estima que en Marzo del 2011 se agotará el total de los bloques /8 libres manejados por el IANA. A partir de dicho momento, los registros regionales no tendrán la posibilidad de solicitar bloques de direcciones adicionales, sólo podrán administrar las direcciones que ya tienen asignadas. La segunda fecha a considerar es cuando los registros agoten su reserva de direcciones y ya no puedan solicitar un bloque adicional al IANA.

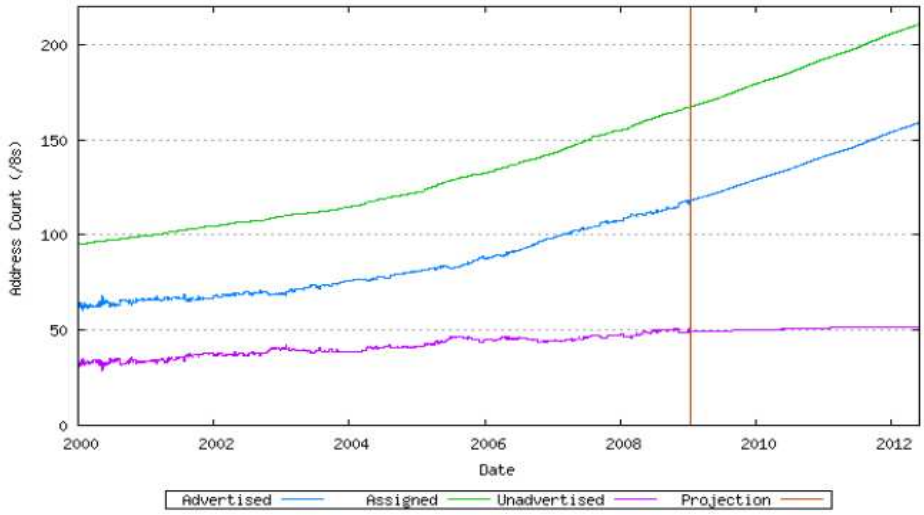


Figura 2-II: Proyección del agotamiento de bloques /8.

Fuente: "IPv4 Address Report".

Se ha estimado que ello ocurra en Mayo del 2012, un año después del agotamiento de los bloques disponibles. Todos estos cálculos y estimaciones están realizados en base al crecimiento histórico que ha tenido la demanda de direcciones IP a nivel mundial. Sin embargo, se espera que en los próximos años, la demanda por direcciones IP sea aún mayor debido a diversos factores tales como:

- ✓ Grandes poblaciones en China, India, Indonesia y África aún no están conectadas.
- ✓ El número de individuos conectados a Internet crece en 77 millones por año.
- ✓ Dispositivos electrónicos de todo tipo están paulatinamente conectándose a Internet.

De todas formas, es posible advertir que en estos días ya estamos en presencia de problemas relacionados con la baja disponibilidad de direcciones IP:

- ✓ Las organizaciones normalmente obtienen pocas direcciones IP para toda su red, limitando las posibilidades de implementar servidores y aplicaciones.
- ✓ Algunos proveedores de servicios (ISP) están asignando direcciones IP privadas a sus subscriptores, lo que significa que el suscriptor no puede ser contactado directamente desde internet.
- ✓ Gran parte de las compañías de telefonía celular no proveen de direcciones públicas a los usuarios de servicios 3G.
- ✓ Muchas aplicaciones disminuyen su rendimiento al no disponer de conectividad punto a punto auténtica.

2.1.2 Problemas de arquitectura

Dado el fuerte crecimiento que ha experimentado Internet en los últimos años, ha sido necesario introducir modificaciones y protocolos complementarios a IPv4, con el fin de poder satisfacer la creciente demanda. Estos cambios han causado que las redes IP estén perdiendo paulatinamente el principio de conectividad punto a punto bajo el cual se diseñó IPv4. Dicho principio estable lo siguiente:

- ✓ Ciertas funciones solo pueden ser realizadas por los nodos finales. El estado de una comunicación punto a punto debe ser mantenida únicamente por los nodos finales y no por la red. La función de la red es enrutar paquetes de forma eficaz y transparente.

- ✓ Los protocolos de transporte están designados para proveer las funciones deseadas sobre una red que no ofrece garantías (mejor esfuerzo).
- ✓ Paquetes deben viajar sin modificación a través de la red.
- ✓ Las direcciones IP son usadas como identificadores únicos para nodos finales.

Una de las medidas introducidas para frenar el agotamiento de direcciones IPv4 es el protocolo de traducción de direcciones de red (NAT). NAT es un protocolo que permite convertir en tiempo real las direcciones utilizadas en los paquetes transportados en una red. El uso de NAT permite que un grupo de dispositivos configurados con direcciones IPv4 privadas compartan un reducido grupo de direcciones IPv4 públicas, permitiendo el acceso hacia Internet. Si bien el uso de NAT ha permitido la expansión actual de Internet, su uso introduce una serie de problemas y desventajas, asociados a la pérdida del principio de conectividad punto a punto. Dentro de las desventajas del uso de NAT podemos encontrar:

- ✓ Complejidad: NAT representa un nivel de complejidad adicional al momento de configurar y manejar una red. Se deben crear grupos de dispositivos y/o redes que comparten un número limitado de direcciones IPv4 públicas.
- ✓ Compatibilidad con ciertas aplicaciones: Muchas aplicaciones no funcionan correctamente cuando se ejecutan desde dispositivos que están en una red donde se realiza NAT. Los desarrolladores han tenido que

inventar nuevos mecanismos para poder funcionar correctamente en dichas redes.

- ✓ Problemas con protocolos de Seguridad: Protocolos de seguridad tales como IPSec están designados para detectar modificaciones en las cabeceras de los paquetes, que es precisamente lo que hace NAT al traducir direcciones. El uso de NAT dificulta la implementación de este tipo de protocolos.
- ✓ Reducción de rendimiento: Por cada paquete que atraviesa una red donde opera NAT, se deben realizar una serie de operaciones adicionales. Dichas operaciones introducen más carga a la CPU del dispositivo que realiza la traducción, disminuyendo su rendimiento.
- ✓ Manejo de estados TCP: El dispositivo que realiza NAT debe manejar y mantener correctamente los estados de cada conexión TCP entre equipos de la red interna y externa.

A pesar de todas sus desventajas, NAT permitió posponer en varios años el agotamiento de direcciones IPv4. Sin embargo, en la actualidad se ha llegado a un punto en donde el uso de NAT no es suficiente para la creciente demanda de direcciones IPv4. Esto ha motivado la evaluación de otras alternativas, tales como IPv6.

2.2 MOTIVADORES DEL CAMBIO A IPV6

El cambio desde IPv4 a IPv6 se suele comparar con la crisis que se vivió a fines de los 90 ante la llegada de año 2000 y sus consecuencias en los sistemas

informáticos. Sin embargo, en el caso de IPv6 no existe una fecha límite o "flag day" en que se puedan deshabilitar todas las redes IPv4 y actualizarlas a IPv6. El proceso de migración debe realizarse en forma progresiva, se prevé que IPv4 siga en funcionamiento durante la próxima década.

El mayor problema que enfrenta IPv6 es que desde el punto de vista de las empresas y organizaciones, su implementación se ve como un gasto poco justificado. En la actualidad, el tráfico IPv6 representa menos de un 1% del tráfico total de Internet, y la mayoría corresponde a Universidades e instituciones que trabajan en el tema. Sin embargo, existen una serie de motivadores para la implementación a IPv6, los que se pueden agrupar en las siguientes categorías.

2.2.1 Motivadores Comerciales

- ✓ La implementación de IPv6 es un movimiento estratégico. Su implementación en las redes de una empresa permite estar preparados para futuras necesidades de los clientes, generando una ventaja comparativa respecto de la competencia.
- ✓ Puede generar un ahorro en los costos de adquisición de nuevos equipos. Diversos fabricantes buscan impulsar la implementación de IPv6, ofreciendo descuentos a empresas e instituciones en la compra de nuevos equipos habilitados para IPv6.
- ✓ Un plan de migración a IPv6 realizado con antelación es más económico que una migración tardía.

- ✓ IPv6 abre las puertas a nuevos productos y servicios a ser ofrecidos por empresas TIC. Sus nuevas características, entre las que destaca el amplio rango de direcciones disponibles, permite generar nuevos proyectos que no podrían ser llevados a cabos en IPv4.

2.2.2 Motivadores Políticos

- ✓ En Estados Unidos, la implementación de IPv6 es un mandato gubernamental, en el que se obligó a todas las agencias a implementar IPv6 en sus redes centrales antes de Junio del 2008. El caso más destacado es el del Departamento de defensa (DOD), el cual realizo un amplio y publicitado plan de integración.
- ✓ Los gobiernos de Japón, China y Corea han establecido la implementación de IPv6 como prioritaria, otorgando un gran apoyo a todas las iniciativas en esta línea. Las olimpiadas de Beijing 2008 fueron un ejemplo de dichas políticas, toda su infraestructura de telecomunicaciones fue implementada mayoritariamente en IPv6.

2.2.3 Motivadores Técnicos

- ✓ Casi la totalidad de los equipos de red, sistemas operativos y dispositivos móviles en venta actualmente proveen soporte para IPv6.
- ✓ El soporte IPv6 que proveen equipos de red como "switches," routers" y "firewalls" ha alcanzado un grado de madurez que ya permite implementar redes que funcionan únicamente con IPv6 sin mayores contratiempos.

- ✓ Algunos ISP ya proveen conectividad IPv6 a usuarios finales.
- ✓ IPv6 facilita la implementación de mecanismos de seguridad y de control de tráfico en redes IP.

En el caso particular de las instituciones de educación superior, como la Escuela Superior Politécnica de Chimborazo, la implementación de IPv6 en sus redes permite además el desarrollo de trabajos de investigación y colaboración en torno a IPv6 y/o a otras tecnologías.

2.3 EL PROTOCOLO IPV6

El protocolo IPv6 comenzó a desarrollarse en el año 1990, tras la primera voz de alerta sobre el posible agotamiento de direcciones IP. Se creó un grupo de trabajo al interior de la IETF, quienes presentaron sus primeras recomendacionessobre el nuevo protocolo que debería reemplazar a IPv4. En el mismo año se publicó oficialmente la primera versión del protocolo IPv6.

En líneas generales, el protocolo IPv6 es considerado una evolución más que una revolución respecto al protocolo IPv4. Se han mantenido los conceptos principales del protocolo, removiendo aquellas características de IPv4 que son poco utilizadas en la práctica. Se han añadido nuevas características que buscan solucionar los problemas existentes en el protocolo IPv4, discutidos en el capítulo 2.1.

2.3.1 Características del protocolo IPv6

Dentro de las principales características de IPv6 se encuentran:

- ✓ Mayor número de direcciones: El tamaño de una dirección aumenta desde

32 a 128[bit] lo que se traduce en alrededor de 3,4·1038 direcciones disponibles. Esto permite asegurar que cada dispositivo conectado a una red pueda contar con una dirección IP pública.

- ✓ Direccionamiento jerárquico: Las direcciones IPv6 globales están diseñadas para crear una infraestructura eficiente, jerárquica y resumida de enrutamiento basada en la existencia de diversos niveles de ISP. Esto permite contar con tablas de enrutamiento más pequeñas y manejables.
- ✓ Nuevo formato de cabecera: Aún cuando el tamaño de la cabecera en IPv6 es mayor que en IPv4, el formato de ella se ha simplificado. Se han eliminado campos que en la práctica eran poco usados, de forma de hacer más eficiente el manejo de los paquetes. Con la incorporación de cabeceras adicionales, IPv6 permite futuras expansiones.
- ✓ Autoconfiguración: IPv6 incorpora un mecanismo de auto configuración de direcciones, "stateless address configuration", mediante el cual los nodos son capaces de auto asignarse una dirección IPv6 sin intervención del usuario.
- ✓ Nuevo protocolo para interactuar con vecinos: El protocolo de descubrimiento de vecinos, reemplaza a los protocolos ARP y "Router Discovery" de IPV4. Una de sus mayores ventajas es que elimina la necesidad de los mensajes del tipo "broadcast".

2.3.2 Estructura de un paquete IPv6

La Figura 2.3 muestra la estructura de un paquete IPv6. Un paquete IPv6 tiene una cabecera de tamaño fijo e igual a 40 [byte], el doble de la cabecera

IPv4. Este aumento se debe a que el tamaño de los campos "Source Address" y "Destination Address" aumentaron su tamaño de 32 a 128 [bit] cada uno.

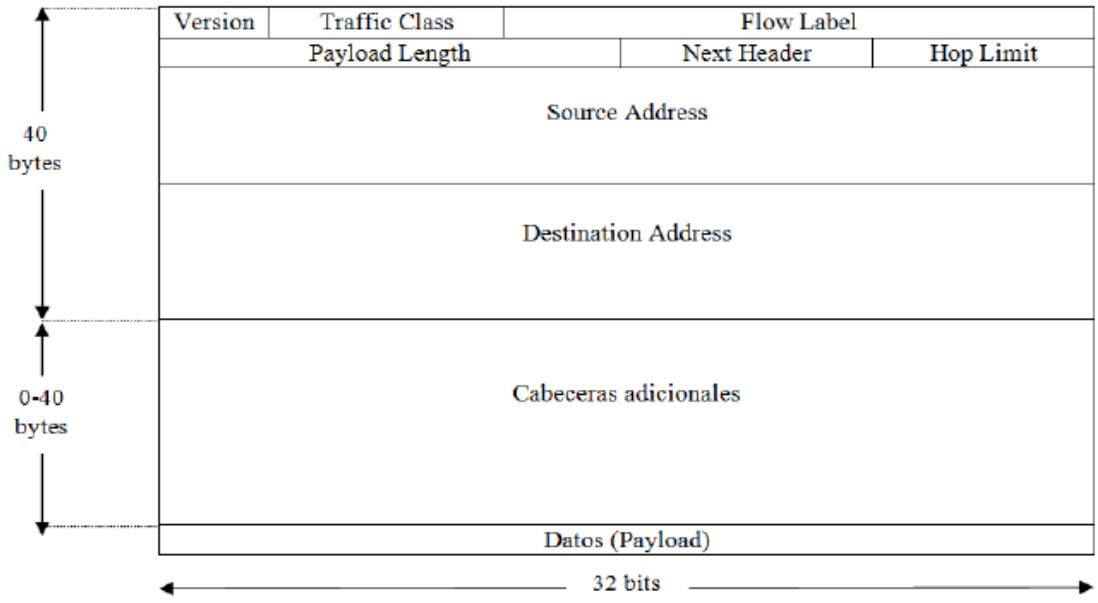


Figura 2-III: Estructura de un paquete IPv6

La cabecera posee los siguientes 8 campos:

Versión ("Version"): Indica la versión del protocolo IP, en este caso su valor es igual a 6.

Clase de tráfico ("Traffic Class"): Incluye información que permite a los "routers" clasificar el tipo de tráfico al que el paquete pertenece, aplicando distintas políticas de enrutamiento según sea el caso. Realiza la misma función que el campo "Type of Service" de IPv4.

Etiqueta de flujo ("Flow Label"): Identifica a un flujo determinado de paquetes, permitiendo a los "routers" identificar rápidamente paquetes que deben ser tratados de la misma manera.

Tamaño de la carga útil ("Payload Length"): Indica el tamaño de la carga útil del paquete. Las cabeceras adicionales son consideradas parte de la carga para este cálculo.

Próximo encabezado ("Next Header"): Indica cual es la siguiente cabecera adicional presente en el paquete. Si no se utilizan, apunta hacia la cabecera del protocolo capa 4 utilizado.

Límite de saltos ("Hop Limit"): Indica el máximo número de saltos que puede realizar el paquete. Este valor es disminuido en uno por cada "router" que reenvía el paquete. Si el valor llega a cero, el paquete es descartado.

Dirección de origen ("Source Address"): Indica la dirección IPv6 del nodo que generó el paquete.

Dirección de destino ("Destination Address"): Indica la dirección de destino final del paquete.

En la Figura 2.4 se pueden apreciar los cambios de la cabecera IPv6 respecto a la cabecera IPv4.

El protocolo IPV6 reemplazó el campo "Options" de IPv4 por las denominadas cabeceras adicionales. Estas cabeceras permiten expandir el funcionamiento de IPv6, sin verse restringidas a un campo de tamaño fijo como el presente en IPv4. Las cabeceras adicionales se ubican inmediatamente después de la cabecera IPv6 y antes de la cabecera del protocolo superior (UDP o TCP).

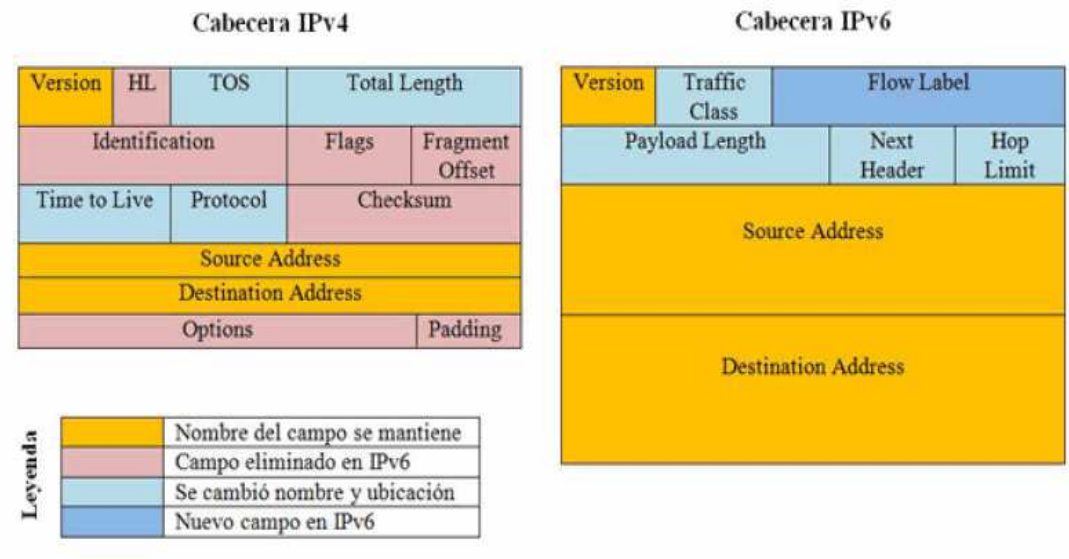


Figura 2-IV: Cambios en la cabecera de los paquetes IPv6.

2.3.3 Formato de una dirección IPv6

Las direcciones IPv6 están compuestas como 8 campos de 16 [bit] de largo, separados por dos puntos ":". Cada campo está representado por 4 caracteres hexadecimales (0-f). Un ejemplo de dirección IPv6 válida es 2001:0000:1234:0000:0000:C1C0:ABCD:0876. Con el fin de simplificar la escritura y memorización de direcciones, se pueden aplicar las siguientes reglas a las direcciones IPv6.

a) No se hace distinción entre mayúsculas y minúsculas. "ABC9"□ es equivalente a "abC9□.

b) Los ceros al inicio de un campo son opcionales. "00c1" es equivalente a "c1".

c) Una sucesión de campos con ceros puede ser reemplazados por "::". "1234:0000:0000:abc9" es igual a „1234::abc9"[2]

Tomando la dirección de ejemplo:

2001:0000:1234:0000:0000:C1C0:ABCD:0876

Mediante la regla a), se puede escribir como:

2001:0000:1234:0000:0000:**c**1**c**0:**abcd**:0876

La dirección se puede escribir de forma resumida utilizando la regla b):

2001:**0**:1234:**0:0**:c1c0:abcd:**876**

Aplicando la regla c) se puede resumir aún más a:

2001:0:1234**::**c1c0:abcd:876

Tal como en el caso de IPv4, para señalar las secciones de la dirección que identifican a la red y al dispositivo, se utiliza el formato CIDR en la forma <dirección>/<prefijo>. Por ejemplo, una dirección en la forma 3ffe:b00:c18:1::1/64 señala que los primeros 64 [bit] identifican a la red (3ffe:b00:c18:1) y los restantes 64[bit] identifican al dispositivo de dicha red (::1).

Tradicionalmente el uso del símbolo ":" en las dirección IPv4 señala un puerto en un determinado nodo, por ejemplo 192.168.1.1:80 señala al puerto 80 (WWW) del nodo 192.168.1.1. Esto representa un problema de incompatibilidad al utilizar direcciones IPv6, por lo que se ha establecido que para señalar un

2Esta regla sólo se puede utilizar una vez en una dirección IPv6, de lo contrario el sistema no sabría cuantos campos se han comprimido en cada caso.

puerto en una determinada dirección IPv6, esta debe estar encerrada por paréntesis cuadrados en la forma [dirección]:puerto, tal como se define en [9].

2.3.4 Algoritmos de Enrutamiento

El uso de IPv6 no implica cambios significativos en la forma en que operan los protocolos de enrutamiento en las redes IP. Sin embargo, para aprovechar las nuevas características de IPv6, se han desarrollado nuevas versiones o complementos a los protocolos de enrutamiento más utilizados. En la Tabla 2.1 se presentan las nuevas versiones desarrolladas para IPv6.

2.3.5 ICMPv6

El protocolo de mensajes de control de Internet (ICMP) es utilizado para enviar información de configuración y reportes de error entre los nodos de una red.Para IPv6, se ha desarrollado una nueva versión del protocolo, denominada ICMPv6 [10]. A diferencia de ICMP para IPv4, el cual no es esencial para lascomunicaciones en redes IPv4, ICMPv6 posee características imprescindiblespara la configuración y comunicación en redes IPv6.

Tabla 2-I: Protocolos de enrutamiento en IPv6

Protocolo enrutamiento	Versión IPv6
RIP	RIPng
EIGRP	EIGRP para IPv6
OSPF	OSPFv3
IS-IS	Integrated IS-IS
BGP	BGP-MP
EIGRP	EIGRP for IPv6

Fuente: LoshinPete IPv6: Theory, Protocol and Practice. Londres, s.n., 2004.

El protocolo ICMPv6 comprende una serie de mensajes, cada uno identificado con un código. Dichos mensajes permiten llevar a cabo diversos procesos en IPv6 tales como: descubrimiento del máximo valor MTU en un camino, manejo de grupos multicast, detección de destinos inalcanzables y el protocolo de descubrimiento de vecinos.

2.3.5.1 Protocolo de descubrimiento de vecinos

NDP es un protocolo necesario para el correcto funcionamiento de las redes IPv6. Es el encargado de descubrir otros nodos en el enlace, realizar la resolución de direcciones IPv6 y direcciones MAC, encontrar los "routers" disponibles y mantener información actualizada sobre el estado de los caminos hacia otros nodos.

Este protocolo realiza funciones para IPv6 similares a las realizadas por ARP en IPV4. Para el intercambio de información, utiliza mensajes ICMPv6. En la Tabla 2.2 Características protocolo descubrimiento de vecinos. Se presentan las funciones que realiza, junto al equivalente en IPv4.

2.3.6 Mecanismos de configuración de direcciones

En IPv6 existen tres distintas formas en las que un nodo puede obtener una dirección IPv6: de forma estática, autoconfiguración sin estados y mediante DHCPv6

2.3.6.1 Configuración estática

La configuración estática consiste en ingresar manualmente la dirección IPv6 de un nodo en un archivo de configuración o mediante el uso de herramientas

propias del sistema operativo. La información que se debe incluir como mínimo es la dirección IPv6 y el tamaño del prefijo de red.

Tabla 2-II: Características protocolo descubrimiento de vecinos.

Característica de NDP	Descripción	Equivalente IPv4
Descubrimiento de "routers"	Permite a los dispositivos detectar a los "routers" presentes en el enlace.	ICMP Router Discovery
Descubrimiento de prefijo	Permite a los nodos conocer el prefijo utilizado en el enlace.	No disponible
Descubrimiento de parámetros	Permite a los nodos auto configurar parámetros como MTU o número máximo de saltos.	PMTU Discovery
Autoconfiguración de direcciones	Permite a los dispositivos auto configurar su propia dirección.	No disponible
Resolución de direcciones	Permite a los nodos determinar las direcciones capa 2 de los dispositivos presentes en el enlace.	ARP
Determinación próximo salto	Permite a los nodos determinar el próximo salto para un destino dado.	Tabla ARP y/o tabla de enrutamiento en los dispositivos.
Detección de vecinos inalcanzables(NUD)	Detecta si se puede alcanzar un determinado nodo.	"Dead Gateway Detection"
Detección de direcciones duplicadas (DAD)	Permite a los nodos determinar si una dirección está en uso.	ARP con origen=0
Redirección	Permite a los "routers" informar a los nodos de un mejor próximo salto para una dirección en particular.	ICMPv4 Redirect

Fuente: LoshinPete *IPv6: Theory, Protocol and Practice*. Londres, s.n., 2004.

2.3.6.2 Autoconfiguración sin estados ("stateless")

El procedimiento de autoconfiguración sin estados utiliza el protocolo de descubrimiento de vecinos NDP para reconocer a los "routers" presentes en el

enlace y generar una dirección IPv6 a partir del prefijo que estos anuncias. Los pasos que realiza un nodo para obtener una dirección son los siguientes:

- ✓ Descubrir un prefijo utilizado en el enlace: El nodo escucha los anuncios que envían los "routers" periódicamente al enlace (mensajes RA) o puede solicitar un anuncio, enviando un mensaje de solicitación de "router" (RS). A partir de los mensajes RA, obtiene la información del prefijo de red.
- ✓ Generar un identificador de interfaz: Para generar el resto de la dirección IPv6, el nodo genera un identificador de interfaz. Puede generarla a partir de su dirección MAC (como en las direcciones locales al enlace) o de forma aleatoria.
- ✓ Verificar que la dirección no esté duplicada: La dirección IPv6 generada debe ser única, por lo que el nodo inicia el procedimiento de detección de direcciones duplicadas (DAD). Si la dirección es única, el nodo comienza a utilizarla.

2.3.6.3 Autoconfiguración con estados (DHCPv6)

La implementación de DHCP para IPv6 (DHCPv6) realiza las mismas funciones que DHCP en IPv4. Un servidor DHCP envía mensajes que contienen la dirección IPv6 a utilizar, dirección del servidor DNS e información adicional a los clientes DHCP, quienes se configuran de acuerdo a la información recibida.

A diferencia de la configuración sin estados, el uso de DHCPv6 permite centralizar toda la asignación de direcciones de los equipo pertenecientes a un sitio. El servidor DHCPv6 no necesita estar conectado en el mismo enlace de

los clientes DHCPv6, los mensajes pueden ser enrutados. En la Tabla 2.3 se observan los principales cambios entre DHCPv4 y DHCPv6.

Tabla 2-III: Diferencias entre DHCPv4 y DHCPv6

Característica	DHCPv4	DHCPv6
Mensaje de reconfiguración	No disponible	Permite a los servidores solicitar a los clientes que actualicen su información.
Dirección de destino de la solicitud DHCP de un cliente	Dirección Broadcast	Grupo multicast que agrupa a todos los servidores DHCP.
Dirección de origen en la solicitud DHCP.	0.0.0.0	Dirección local de enlace del nodo.
Asociación de identidad	No disponible	Los clientes pueden solicitar información a varios servidores DHCPv6 y obtener múltiples direcciones.
Etiqueta de configuración asistida	No disponible	Un "router" puede anunciar a los nodos si es que está permitido el uso de DHCPv6.

Fuente: LoshinPete *IPv6: Theory, Protocol and Practice*. Londres, s.n., 2004.

2.4 VOIP[3]

Las señales digitales han prevalecido sobre las analógicas puesto que ofrecen mayores ventajas entre las que se pueden resaltar: Facilidad para multicanalizar las señales, fácil señalización, generación de señales, baja razón señal-ruido y una encriptación eficiente de la señal, la cual importa mucho en las comunicaciones militares y cualquier otra que requiera cumplir con niveles buenos de seguridad. La red IP comenzó a desarrollarse exponencialmente con el surgimiento del Internet. Surgieron los conceptos de nodos, servidores, enrutadores, repetidores, puentes, switches, gateways y demás elementos que conforman una red de paquetes conmutados para el intercambio de datos.

3Switching to VoIP.By Theodore Wallingford

Poco a poco la información que se buscaba transmitir empezó a ser más demandante, al grado de aplicaciones populares como un Chat que no sólo comunica a dos usuarios por medio de mensajes escritos en tiempo real, sino que también les otorgaba la oportunidad deestablecer una conversación oral y visual con sólo una PC, micrófono, bocinas, cámara web y una conexión a Internet. Llegó el momento en el que por la red viajaban datos multimedia como videoconferencias a una tasa alta de transmisión y muestran una fuerte evolución en las comunicaciones digitales. Es así como surgió la idea de implementar una red IP donde pudiera viajar la voz. Se ha preferido la red de paquetes conmutados sobre la red de circuitos conmutados puesto que la segunda exige un ancho de banda definido o fijo durante toda la transmisión punto a punto incluso cuando no se esté utilizando por completo este recurso, por ejemplo cuando ambas personas guardan silencio por instantes. Todo lo contrario ocurre en la red de paquetes conmutados, donde el ancho de banda es aprovechado al máximo.

Lo anterior se puede traducir en la diferencia de costos invertidos en cada red. Un objetivo de voz sobre IP es unificar las redes de voz y las de datos, de esta forma se adquieren muchos beneficios.

2.4.1 Estructura de la red VoIP.

La estructura de la red de voz sobre IP es la misma estructura que se maneja en Internet, las aplicaciones, los medios de transporte, la organización del ruteo sobre la red, los modos de enlace y la transmisión de la señal por los medios físicosforman parte del modelo OSI. La ventaja de la red VoIP es que

no importa el tipo de aplicación mientras ésta pueda transformar su información en datos, segmentos, paquetes, tramas y finalmente bits.

El protocolo que se utiliza para la capa de transporte es el RTP (Real-time Transfer Protocol) en segmentos de tipo UDP sobre paquetes IP. Se ha escogido éste sobre el TCP dado que, TCP es caracterizado por ser un protocolo donde se deben recibir señales de reconocimiento (acknowledge) por parte del receptor antes de enviar el siguiente segmento, es decir es un protocolo orientado a conexión que ofrece seguridad a la transmisión y recepción de los paquetes aunque introduce retardos en la comunicación.

El concepto de conmutador (central local, central de grupo, etc. Para una red tradicional de conmutación de circuitos) en VoIP es el Media Gateway Controller (MGC). Éste es un conjunto de productos, protocolos y aplicaciones capaces de permitir que cualquier dispositivo acceda a los servicios de Internet y de Telecomunicaciones sobre las redes IP. Este elemento es la pieza central en la red de telefonía IP, ya que es capaz de manejar inteligentemente las llamadas en la plataforma de servicio de los Proveedores de Servicio de Internet (ISP, Internet Service Provider). Por otro lado, sirven como plataformas de integración para aplicaciones e intercambio de servicios y son capaces de transportar tráfico de voz, datos y video de una manera más eficiente que los equipos existentes.

El trabajo dentro del Media Gateway Controller es realizado por medio de hardware y software inteligentes; denominados por algunos autores como Softswitch y Gatekeeper (para el caso de redes H.323).

En sí, estos tres elementos forman parte del mismo sistema, en otras palabras, el Gatekeeper es el hardware, el SoftSwitch es el software y ambos son controlados porel Media Gateway Controller. Todas las tareas se pueden dividir en cinco secciones:

Gateway Controller, Media Gateway, Signaling Gateway, Media Server y Feature Server. El Media Gateway Controller es eficiente gracias a su interacción con el Media Gateway y el Signaling Gateway. Las funciones principales son: control de llamada, protocolos de establecimiento de llamadas como H.323 y SIP, protocolos de control de media por ejemplo MGCP y H.248, control sobre la calidad y clase de servicio, conocimiento del enrutamiento, plan de numeración local, detalle de las llamadas para facturación, control de manejo del ancho de banda, crear un puente entre la señalización SS7 y VoIP, entre muchas otras más. Un ejemplo de elemento utilizado en redes VoIP es el puerto FXO que permiten conectar directamente una línea privada de una compañía a la PSTN, posibilitando a los terminales IP hacer llamadas a cualquier teléfono análogo. Con este "Gateway" se pueden realizar llamadas hacia y desde terminales telefónicos que no tienen acceso a internet. Los puertos FXS (Foreign Exchange Station) conectan su teléfono o fax convencional a la red VoIP. Se puede marcar hacia el exterior a través de un Gateway a otras Gateways o Teléfonos IP. La Figura 2.4.1 muestra la estructura general de la red VoIP.

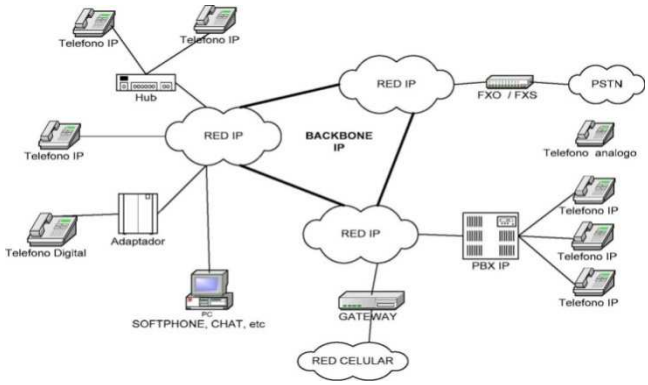


Figura 2-V: Estructura general de una red VoIP

2.4.2 Codecs.

La red VoIP no sería posible sin que se realizara un proceso de compresión y descompresión de Voz, donde primero es codificada desde su estado análogo digital en paquetes IP, que pueden ser enviados a través de la red; finalmente se decodifican a su estado análogo original, es decir nuevamente a voz, en el terminal receptor. Para aplicaciones VoIP los más populares y/o utilizados son: G.711, G.723.1, y el G.729, Además de la ejecución de la conversión de analógico a digital, el CODEC comprime la secuencia de datos, y proporciona la cancelación del eco. La compresión de la forma de onda representada puede permitir el ahorro del ancho de banda. Esto es especialmente interesante en los enlaces de poca capacidad y permite tener un mayor número de conexiones de VoIP simultáneamente.

2.4.3 Ventajas de la VoIP.

Resulta fácil enumerar las siguientes ventajas:

- ✓ Ahorro en los costos de Administración. Todos los dispositivos telefónicos aprovechan el cableado Ethernet existente, con lo que se simplifica la

instalación y mantenimiento del sistema telefónico.

- ✓ Permite la integración de aplicaciones propias de una institución o compañía, tales como los sistemas CRM (Customer Relationship Management) y de Centros de contacto (Contact Center).
- ✓ Se tienen acceso a servicios adicionales, tales como: mensajería unificada, administración de llamadas entrantes y salientes, control del flujo telefónico, etc..
- ✓ Mayor funcionalidad. Integración total con los sistemas PC actuales. Un ejemplo de esta integración es que podemos marcar el teléfono al que queremos llamar directamente desde Outlook.
- ✓ Escalable, las PBX convencionales tienen capacidades fijas que al ser sobrepasadas requieren el cambio completo del sistema, esto no es el caso con VoIP.
- ✓ Mejora la productividad. VoIP trata a la voz como si fuera cualquier otro tipo de dato, así los usuarios pueden adjuntar documentos a los mensajes de voz o participar en reuniones virtuales usando datos compartidos y video conferencias.

2.4.4 Calidad de la voz

Con la migración de la tecnología de conmutación de circuitos a la conmutación de paquetes IP en el proceso transmisión de voz, se introducen nuevas consideraciones, tales como pérdida de paquetes (lost), retardo de paquetes (delay) y el desplazamiento en el tiempo de los paquetes (jitter).

Adicionalmente, problemas antiguos, tales como el eco, variación en el nivel de saturación y ruido de fondo, que eran problemas de los sistemas de conmutación de circuitos, también están presentes en las redes VoIP. La combinación de todos estos problemas que afectan la calidad de voz, se presentan como un gran reto para los planificadores y diseñadores de soluciones de comunicación VoIP.

2.4.4.1 Factores que influyen en la calidad de voz.

Debido a que la telefonía es un servicio orientado al cliente, los parámetros de medición de la calidad de la voz están basados en la apreciación que los usuarios tienen, sobre todo cuando se trata de una aplicación en tiempo real, como lo es una conversación telefónica. Por lo tanto la meta principal en la medición y evaluación de la calidad de voz en redes conmutadas por paquetes es el desarrollo de indicadores de la percepción que el usuario tiene de la calidad de voz, que sean confiables y creíbles, de tal forma que reflejen los efectos específicos de la conmutación de paquetes.

2.4.4.2 Consideración del usuario sobre la calidad de la Voz.

Cuando los usuarios hablan acerca de la calidad de voz, ellos tratan de describir generalmente su reacción a, o su insatisfacción con, uno de los dos siguientes atributos:

- ✓ Calidad de Conexión. Determinada por lo que es escuchado sobre la conexión.

Usabilidad de la conexión. Determinada por lo que se experimenta en los intercambios de conversación sobre la conexión.

En términos simples uno de los métodos para poder medir la calidad de la voz se realiza a través de encuestas, en las cuales el usuario califica diferentes factores del servicio de voz que utiliza, lo que lo convierte en una estimación subjetiva de la calidad de la voz.

Por otro lado algunos investigadores han intentado evaluar la calidad de la voz en forma objetiva, usando una variedad de mediciones espectrales, mediciones de ruido y mediciones paramétricas. Ambas formas de determinar la calidad de la voz, es decir la subjetiva y la objetiva tienen sus pros y sus contras. Debido a todas las variables consideradas, se necesitan en el momento de diseño de una solución de comunicaciones el empleo de herramientas que permitan garantizar que estos factores se lleven a niveles aceptables para los usuarios; tales herramientas pueden ser protocolos y características propias de la red que garanticen mejorar la calidad de servicio (QoS), para lo cual deben hacerse análisis del tráfico de la red como mínima opción, y finalmente no se descarta utilizar herramientas en forma de software desarrollado especialmente para poder medir los niveles en la calidad de servicio y de voz en una red.

2.5 PROTOCOLOS UTILIZADOS EN VOIP

Figura 2-VI: Protocolos VoIP

2.5.1 SIP

[4]SIP son las siglas en inglés del Protocolo para Inicio de Sesión, siendo un estándar desarrollado por el IETF, identificado como RFC 3261, en 2002. SIP es un protocolo de señalización para establecer las llamadas y conferencias en redes IP. El inicio de la sesión, cambio o término de la misma, son independientes del tipo de medio o aplicación que se estará usando en la llamada; una sesión puede incluir varios tipos de datos, incluyendo audio, video y muchos otros formatos. Es un protocolo de control que se encuentra en la capa de aplicación del modelo OSI para crear, modificar y terminar sesiones con uno o más participantes. Las sesiones incluyen: llamadas telefónicas, transferencias de datos multimedia, y conferencias en tiempo real.

Las invitaciones SIP usadas para crear sesiones, llevan consigo la descripción de la sesión y esto permite a los participantes buscar la compatibilidad. SIP utiliza elementos llamados servidores proxy para ayudar a enrutar las peticiones a los usuarios de una zona, autentificar y autorizar servicios para

4 RFC 3261 SIP: Session Initiation Protocol

éstos, e implementar políticas para el ruteo de llamadas. SIP puede viajar sobre cualquier protocolo de transporte. Los usuarios son denominados user agent y éstos se pueden desplazar a través de la red y obtener diferentes denominaciones y mandar diversos tipos de datos (voz, texto, video). SIP ofrece la ventaja de invitar a los nuevos participantes a la sesión creando una nueva infraestructura en donde todos los user agent pueden registrarse, invitar a nuevas sesiones, modificar las características de la sesión, etc. A pesar de la movilidad del usuario, su identificador puede ser permanente sin importar la red en la que se encuentre. SIP posee las siguientes funciones principales:

- ✓ Determina los tipos de hosts que pretenden establecer una comunicación.
- ✓ Determina la disponibilidad de la persona que recibe la llamada para conectarse.
- ✓ Determina el tipo de datos y sus parámetros necesarios que se usarán durante la comunicación.
- ✓ Establece los parámetros de la sesión tanto en la persona que llama como en la que es llamada.
- ✓ Administra la sesión, en otras palabras, inicialización, transferencia, modificación y terminación de sesiones.

SIP es un protocolo el cual no trabaja de manera única sino que, lo hace en conjunto con otros protocolos de la IETF para crear una arquitectura multimedia más completa. Estos otros protocolos son: el RTP (Real-time Transport Protocol) para el envío de datos y revisar la calidad del servicio, RTSP (Real Time Streaming Protocol) para controlar el envío de datos multimedia,

MEGACO (Media Gateway Control) para controlar las conmutaciones con la red PSTN y el SDP (Session Description Protocol), Protocolo de descripción de sesión para la descripción de las diferentes sesiones. SIP funciona tanto con IPv4 como IPv6. El propósito de SIP es la comunicación entre dispositivos multimedia. SIP hace posible esta comunicación gracias a dos protocolos que son RTP/RTCP y SDP. El protocolo RTP se usa para transportar los datos de voz en tiempo real (igual que para el protocolo H.323, mientras que el protocolo SDP se usa para la negociación de las capacidades de los participantes, tipo de codificación, etc.). SIP fue diseñado de acuerdo al modelo de Internet. Es un protocolo de señalización extremo a extremo que implica que toda la lógica es almacenada en los dispositivos finales (salvo el ruteado de los mensajes SIP). El estado de la conexión es también almacenado en los dispositivos finales. El precio a pagar por esta capacidad de distribución y su gran escalabilidad es una sobrecarga en la cabecera de los mensajes producto de tener que mandar toda la información entre los dispositivos finales. SIP es un protocolo de señalización a nivel de aplicación para establecimiento y gestión de sesiones con múltiples participantes. Se basa en mensajes de petición y respuesta y reutiliza muchos conceptos de estándares anteriores como HTTP y SMTP. SIP soporta funcionalidades para el establecimiento y finalización de las sesiones multimedia: localización, disponibilidad, utilización de recursos, y características de negociación. Para implementar estas funcionalidades, existen varios componentes distintos en SIP. Existen dos elementos fundamentales, los agentes de usuario (UA) y los servidores. User Agent (UA): consisten en dos partes distintas, el User Agent

Client (UAC) y el User Agent Server (UAS). Un UAC es una entidad lógica que genera peticiones SIP y recibe respuestas a esas peticiones. Un UAS es una entidad lógica que genera respuestas a las peticiones SIP. Ambos se encuentran en todos los agentes de usuario, así permiten la comunicación entre diferentes agentes de usuario mediante comunicaciones de tipo cliente-servidor. Los servidores SIP pueden ser de tres tipos:

- ✓ Proxy Server: retransmiten solicitudes y deciden a qué otro servidor deben remitir, alterando los campos de la solicitud en caso necesario. Es una entidad intermedia que actúa como cliente y servidor con el propósito de establecer llamadas entre los usuarios. Este servidor tienen una funcionalidad semejante a la de un Proxy HTTP que tiene una tarea de encaminar las peticiones que recibe de otras entidades más próximas al destinatario. Existen dos tipos de Proxy Servers: Statefull Proxy y Stateless Proxy.
    - o Statefull Proxy: mantienen el estado de las transacciones durante el procesamiento de las peticiones. Permite división de una petición en varias (forking), con la finalidad de la localización en paralelo de la llamada y obtener la mejor respuesta para enviarla al usuario que realizó la llamada.
    - o Stateless Proxy: no mantienen el estado de las transacciones durante el procesamiento de las peticiones, únicamente reenvían mensajes.
- ✓ Register Server: es un servidor que acepta peticiones de registro de los

usuarios y guarda la información de estas peticiones para suministrar un servicio de localización y traducción de direcciones en el dominio que controla.

- ✓ Redirect Server: es un servidor que genera respuestas de redirección a las peticiones que recibe. Este servidor reencamina las peticiones hacia el próximo servidor. La división de estos servidores es conceptual, cualquiera de ellos puede estar físicamente en una única máquina, la división de éstos puede ser por motivos de escalabilidad y rendimiento.

2.5.1.1 Mensajes y respuestas SIP.

SIP utiliza Métodos / Solicitudes y correspondientes Respuestas para establecer una sesión de llamada. SIP además es un protocolo textual que usa una semántica semejante a la del protocolo HTTP. Los UAC realizan las peticiones y los UAS retornan respuestas a las peticiones de los clientes. SIP define la comunicación a través de dos tipos de mensajes. Las solicitudes (métodos) y las respuestas (códigos de estado) emplean el formato de mensaje genérico establecido en el RFC 2822 , que consiste en una línea inicial seguida de uno o más campos de cabecera (headers), una línea vacía que indica el final de las cabeceras, y por último, el cuerpo del mensaje que es opcional.

Las peticiones SIP son caracterizadas por la línea inicial del mensaje, llamada Request-Line, que contiene el nombre del método, el identificador del destinatario de la petición (Request-URI) y la versión del protocolo SIP. Existen seis métodos básicos SIP (definidos en RFC 254) que describen las peticiones de los clientes:

- ✓ **INVITE:** Este método indica que el usuario o servicio es invitado a participar en una sesión. Incluye una descripción de sesión y, para llamadas de full dúplex, la parte llamante indica el tipo de medio. Una respuesta con éxito a una invitación INVITE de dos partes (respuesta 200 OK) incluye el tipo de medios recibidos por la parte llamada. Con este simple método, los usuarios pueden reconocer las posibilidades del otro extremo y abrir una sesión de conversación con un número limitado de mensajes e idas y vueltas.
- ✓ **ACK:** Estas respuestas corresponden a una petición INVITE. Representan la confirmación final por parte del sistema final y concluye la transacción indicada por el comando INVITE. Si la parte llamante incluye una descripción de la sesión, los parámetro en la petición INVITE se utilizan como los predeterminados.
- ✓ **OPTIONS:**Este método permite consultar y reunir posibilidades de agentes de usuarios y servidores de red. Sin embargo, esta petición no se utiliza para establecer sesiones.
- ✓ **BYE:** Este método se utiliza por las partes que llaman y son llamadas para liberar una llamada. Antes de liberar realmente la llamada, el agente de usuario envía esta petición al servidor indicando el deseo de terminar la sesión.
- ✓ **CANCEL:** Esta petición permite que los agentes de usuario y servidores de red cancelen cualquier petición que este en progreso. Esto no afecta a las peticiones terminadas en las que las respuestas finales ya fueron recibidas.
- ✓ **REGISTER:** Ese método se utiliza por los clientes para registrar información de

localización con los servidores SIP. Sin embargo, existen otros métodos adicionales que pueden ser utilizados, publicados en otros RFCs como los métodos INFO, SUBSCRIBER, etc.

2.5.1.2 Respuestas (Códigos de estado) SIP.

Después de la recepción e interpretación de un mensaje de solicitud SIP, el receptor del mismo responde con un mensaje. Este mensaje, es similar al anterior, difiriendo en la línea inicial, llamada Status-Line, que contiene la versión de SIP, el código de la respuesta (Status–Code) y una pequeña descripción (Reason-Phrase). El código de la respuesta está compuesto por tres dígitos que permiten clasificar los diferentes tipos existentes. El primer dígitodefine la clase de la respuesta.

Una de las funciones de los servidores SIP es la localización de los usuarios y resolución de nombres. Normalmente, el agente de usuario no conoce la dirección IP del destinatario de la llamada, sino su e-mail. Las entidades SIP identifican a un usuario con las SIP URI (Uniform Resource Identifiers) definido en el RFC 2396. Una SIP URI tiene un formato similar al del email, consta de un usuario y un dominio delimitado por una @, como muestra los siguientes casos:

- usuario@dominio, donde dominio es un nombre de dominio completo.
- usuario@equipo, donde equipo es el nombre de la máquina.
- usuario@dirección_ip, donde dirección_ip es la dirección IP del dispositivo.
- número_teléfono@gateway, donde el gateway permite acceder al

número de teléfono a través de la red telefónica pública.

La solución de identificación de SIP, también puede ser basada en el DNS descrito en el RFC 3263, donde se describen los procedimientos DNS utilizados por los clientes para traducir una SIP URI en una dirección IP, puerta y protocolo de transporte utilizado, o por los servidores para retornar una respuesta al cliente en caso de que la petición falle

Tabla 2-IV: Clases de código de estado

Clase de respuesta	Código de estado	Explicación
Informativa	100	Tratando
	180	Sonando
	181	La llamada esta siendo reenviada
	182	Puesta en cola
	183	Progreso de sesión
Success	200	OK
	300	Elección múltiple
	301	Movida permanente
	302	Movida temporalmente
	303	Véase otra
	305	Utilizar Proxy
	380	Servicio alternativo
Errores de solicitud	400	Petición defectuosa
	401	No autorizado
	402	Se requiere pago
	403	Prohibido

	404	No encontrado
	405	Método no permitido
	406	No aceptable
	407	Se requiere autenticación de proxy
	408	Se acaba tiempo de petición
	409	Conflicto
	410	Se ha marchado
	411	Se requiere longitud
	413	Entidad pedida demasiado larga
	414	URL pedido demasiado largo
	415	Tipo de medio no soportado
	420	Extensión errónea
	480	No disponible temporalmente
	481	Segmento de llamada o transacción no existe
	482	Detectado bucle
	483	Demasiados saltos
	484	Dirección incompleta
	485	Ambiguo
	486	ocupado
Errores de servidor	500	Error interno de servidor
	501	Sin implementar
	502	Gateway erróneo
	503	Servicio no disponible
	504	Gateway fuera de tiempo
	505	Versión SIP no soportada
	600	Ocupado en todos partes
	603	Rechazado
	604	No existe en ningún sitio
	606	No aceptable

Fuente: http:// www.ipv6.org/

El protocolo SDP (Session Description Protocol) RFC 2327 se utiliza para describir sesiones multicast en tiempo real, siendo útil para invitaciones, anuncios, y cualquier otra forma de inicio de sesiones. La propuesta original de SDP fue diseñada para anunciar información necesaria para los participantes y para aplicaciones de multicast MBONE (Multicast Backbone). Actualmente, su uso está extendido para el anuncio y la negociación de las capacidades de una sesión multimedia en Internet.

Puesto que SDP es un protocolo de descripción, los mensajes SDP se pueden transportar mediante distintos protocolos con SIP, RTSP, correo electrónico con

aplicaciones MIME o protocolos como HTTP. Como el SIP, el SDP utiliza la codificación del texto. Un mensaje del SDP se compone de una serie de líneas, denominados campos, dónde los nombres son abreviados por una sola letra, y está en una orden requerida para simplificar el análisis. El SDP no fue diseñado para ser fácilmente extensible. A continuación se analizará una llamada. En una llamada SIP hay varias transacciones SIP. Una transacción SIP se realiza mediante un intercambio de mensajes entre un cliente y un servidor. Consta de varias peticiones que se muestran en la Figura 2.7

Las dos primeras transacciones corresponden al registro de los usuarios. Los usuarios deben registrarse para poder ser encontrados por otros usuarios. En este caso, los terminales envían una petición REGISTER, donde los campos from y to corresponden al usuario registrado

El servidor Proxy, que actúa como Register, consulta si el usuario puede ser autenticado y envía un mensaje de OK en caso positivo. La siguiente transacción corresponde a un establecimiento de sesión.

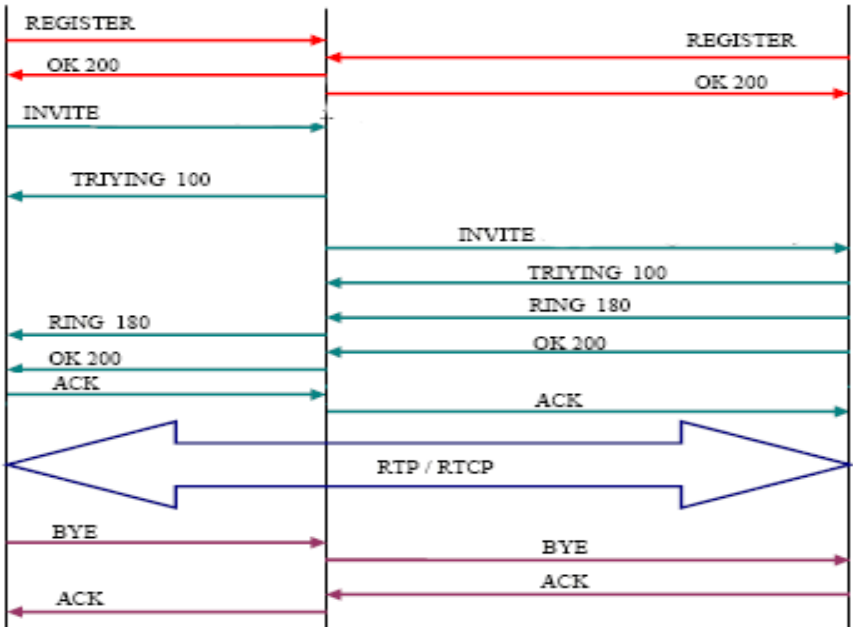


Figura 2-VII: Mensajes SIP para una llamada.

Esta sesión consiste en una petición INVITE del usuario al proxy. Inmediatamente, el proxy envía un TRYING 100 para parar las retransmisiones y reenvía la petición al usuario B. El usuario B envía un Ringing 180 cuando el teléfono empieza a sonar y también es reenviado por el proxy hacia el usuario A. Por último, el OK 200 corresponde a aceptar la llamada (el usuario B descuelga). En este momento la llamada está establecida, pasa a funcionar el protocolo de transporte RTP con los parámetros (puertos, direcciones, codecs, etc.) establecidos en la negociación mediante el protocolo SDP. La última transacción corresponde a una finalización de sesión. Esta finalización se lleva a cabo con una única petición BYE enviada al Proxy, y posteriormente reenviada al usuario B. Este usuario contesta con un OK 200 para confirmar que se ha recibido el mensaje final correctamente.

SIP es parte de los estándares de IETF y se modela en otros protocolos delnternet, tales como SMTP y HTTP. Se utiliza para establecer y cambiar entreunos o más usuarios en una red IP. En la Figura 2.8 se muestra las capas delprotocolo SIP.

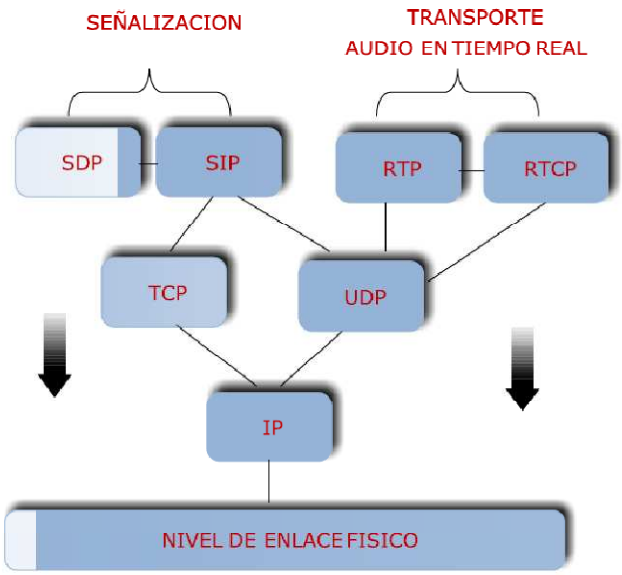


Figura 2-VIII: Capas del Protocolo SIP

2.5.2 RTP

RTP es el más popular de los protocolos de transporte de VoIP. Seespecifica en el RFC 1889 bajo el título de "RTP: Un protocolo de transporte paraaplicaciones en tiempo real." Este RFC describe RTP y RTCP. Pues los nombressugerirían que estos dos protocolos son necesarios para soportar aplicaciones entiempo real como voz y vídeo. RTP funciona sobre la capa de UDP, que no evitapérdida de paquetes ni garantiza el orden correcto para la entrega de paquetes.

Los paquetes de RTP superan esos defectos incluyendo los números de serie queayudan a RTP para detectar los paquetes perdidos y para asegurar la

entrega delpaquete en el orden correcto. Los paquetes de RTP incluyen una etiqueta defecha/hora, donde indica el tiempo en que el paquete se envió de la fuente.

Esta etiqueta de fecha/hora ayuda con la sincronización del usuario dedestino, en donde se calcula el retraso que se tiene y el jitter, dos detractores muyimportantes de calidad de la voz. RTP no tiene la capacidad de corregir el retrasoni el jitter, pero proporciona la información adicional a un uso más alto de la capa,de modo que pueda hacer determinaciones en cuanto a cómo un paquete de vozo de datos se maneja lo mejor posible.

RTCP proporciona un número de mensajes que se intercambian entre losusuarios de sesión y que proporcionan la regeneración en la sesión. El tipo deinformación incluye los detalles, tales como los números de los paquetes perdidosde RTP, retraso y jitter entre llegadas. Mientras que los paquetes de voz setransportan en paquetes de RTP, los paquetes de RTCP transfieren laregeneración de la calidad. Siempre que una sesión de RTP se abra, una sesión

de RTCP también se abre, es decir, cuando un número de acceso de UDP seasigna a una sesión de RTP para la transferencia de los paquetes de los medios,otro número de acceso se asigna para los mensajes de RTCP. En la figura 2.9 semuestran la pila de RTP.

Figura 2-IX: Protocolo RTP

2.5.2.1 RTCP

RTCP permite intercambios de información de control entre losparticipantes de la sesión con el fin de proporcionar la regeneración de la calidad.Esta regeneración se utiliza para detectar y para corregir problemas en ladistribución. La combinación del multicast de RTCP y de IP permite al operador dela red monitorear la calidad. RTCP proporciona la información en la calidad de unasesión de RTP. RTCP autoriza a los operadores de la red para obtener lainformación sobre el retraso, el Jitter y la pérdida de paquetes, así tomar la accióncorrectiva para mejorar la calidad.

2.5.3 Software utilizado en telefonía IP.

Es válido hacer mención, que cuando se tiene el reto de crear soluciones tecnológicas, cuya implementación es una mezcla de Hardware y Software, se tienen dos opciones técnicamente válidas: Hardware y Software propietario o Hardware y Software Libre (Open Source). Las ventajas de usar Software Libre, no se remiten únicamente al aspecto económico, ya que a diferencia del Software gratuito, el software libre ofrece siempre los archivos fuente, que le permiten a cualquier implementador de soluciones, usarlos de tal forma que

puede adecuar esa aplicación a las necesidades específicas existentes. Una de las características más importantes de la implementación del Laboratorio para el Estudio de protocolos de VoIP, es que estará basado en software Libre. Existe una gran variedad de Software GPL o libre y se han seleccionado como herramientas de trabajo los más sobresalientes por su versatilidad, compatibilidad y desempeño.

2.5.4 Factores que determinan la calidad de la voz en sistemas VoIP

2.5.4.1 Latencia o Retardo

Una red Ethernet cableada o inalámbrica no fue diseñada para aplicaciones en tiempo real o con una garantía en la entrega de paquetes. La congestión de la red inalámbrica, sin diferenciación del tráfico, puede rápidamente hacer la voz inutilizable. Al procesamiento de la señal de voz en los puntos de envió y recepción, incluyendo el tiempo necesario para codificar o decodificar la señal de voz analógica o digital en el sistema de codificación de voz elegido, se le suma al retraso. La compresión de la señal de voz también aumentará el retraso, entre mayor sea la compresión mayor será el retraso. En caso de que los costos de ancho de banda no sean una preocupación, un prestador de servicios puede utilizar el códec G.711, que tiene una velocidad de descompresión de (64 Kbps), que representa un mínimo de retraso debido a la compresión.

En la parte de la transmisión, el retardo por paquetización es otro factor que debe tomarse en cuenta. El retraso de paquetización es el tiempo que tarda para formarse un paquete con los datos, cuanto mayor sea el tamaño del

paquete se necesita más tiempo. El uso de tamaños de paquetes más cortos pueden reducir este retraso, pero esto provocará que se incremente la actividad en la red porque más paquetes han de ser enviados, y todos contendrán información similar en la cabecera. El equilibrio entre calidad de voz, el retraso por paquetización y el uso eficiente del ancho de banda son muy importantes a la hora de proveer un servicio de VoIP.

¿Cuánta retraso puede ser demasiado? De todos los factores que degradan las comunicaciones VoIP, el retardo es el mayor. La latencia de menos de 100 ms no afecta la voz. Sin embargo, la latencia superior a 120 ms es discernible para la mayoría, y en 150 ms la calidad de voz ha disminuido de forma notable. Un desafío para los futuros proveedores de servicios de VoIP es obtener una latencia de cualquier conversación en su red, que no sobrepase los 100 ms. Los seres humanos son perceptibles a los retrasos de más de alrededor de 200 ms. La ITU-T G.114 especifica que el retraso no debe ser superior a 150 ms en un sentido del envió de la información ó 300 ms de ida y vuelta. El dilema es que, si bien aplicaciones por ejemplo (correo electrónico,) pueden tolerar una cantidad de retraso, por lo general estas aplicaciones tratan de consumir cada bit de la capacidad de la red que pueden. En contraste las aplicaciones de voz sólo necesitan pequeñas cantidades de la red, pero esa suma tiene que estar disponible en un momento inmediato.

El retraso experimentado por una llamada se produce en el lado de la transmisión, en la red y en lado de la recepción. La mayor parte del retraso en el lado de la transmisión es debido al retraso producido por el códec. En la

red, la mayor parte del retraso se debe al tiempo de la transmisión (señalización y propagación) y el tiempo en las colas del ruteador. Por último, el jitter, el procesamiento y en algunas implementaciones añaden retraso en el lado de la recepción.

El retraso introducido por el codificador de voz puede dividirse en algorítmico y el retraso de procesamiento. El algoritmo de retraso se produce debido a la elaboración del bloque de procesamiento, ya que el codificador produce un conjunto de bits que representan un bloque de muestras de voz.

2.5.4.2 Paquetes perdidos.

En redes, un porcentaje de los paquetes pueden perderse o retrasarse, especialmente durante los períodos de congestión. Asimismo, algunos paquetes son descartados debido a errores durante la transmisión. Paquetes perdidos, retrasados, dañados y deteriorados, se ve reflejado en la calidad de voz.

En técnicas convencionales de corrección de errores utilizadas en otros protocolos, los bloques de datos que contengan errores se descartan, y los que recibe la computadora solicitan la retransmisión del paquete. De este modo el mensaje que es finalmente entregado al usuario no es exactamente el mismo mensaje que se originó. Porque sistemas VoIP son sensibles al tiempo y no pueden esperar para la retransmisión, los sistemas más sofisticados de detección y corrección de errores utilizan sonido para llenar huecos en las llamadas. Este proceso es una parte de la voz del emisor y luego utilizando un complejo algoritmo para aproximar el contenido de los paquetes que faltan, el

nuevo sonido de información es creado para mejorar la comunicación. De este modo, el sonido escuchado por el receptor no es exactamente el sonido de transmisión, sino más bien parte de los que han sido creados por el sistema para mejorar el sonido emitido.

La mayoría de las pérdidas de los paquetes se producen en los ruteadores, ya sea debido a las altas transferencias de carga o alta carga de enlace. En ambas situaciones, los paquetes en las colas podrían ser eliminados. Otra fuente de pérdida de paquetes son los errores en los enlaces de transmisión.

La configuración de errores y colisiones podrían también generar pérdidas de paquetes. En aplicaciones de tiempo no real, las pérdidas de paquetes seresuelven en la capa del protocolo de transmisión (TCP). Para la telefonía esto noes una solución viable ya que se volvería a transmitir los paquetes que llegandemasiado tarde y no sería de mucha utilidad.

Tal vez el principal desafío para VoIP es que en relación con las redescableadas, los paquetes se reducen en una tasa excesiva (más de 30%). Estopuede conducir a la distorsión de la voz en la medida en que la conversación vasiendo difícil. En pasarelas de VoIP diseñados para redes de cable, una soluciónes usar un buffer de jitter.

2.5.4.3 Jitter

El jitter es un efecto de las redes de datos no orientadas a conexión y basadas en conmutación de paquetes. Como la información se discretiza en paquetes cada uno de los paquetes puede seguir una ruta distinta para llegar al destino. 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". Las comunicaciones en tiempo real (como VoIP) son especialmente sensibles a este efecto.

2.5.4.4 MOS[5]

Laindustria telefónica emplea un sistema de calificación subjetiva conocido comoMOS, para medir la calidad de sus conexiones telefónicas. Las técnicas demedición se definen en el ITU-T P.800 y se basan en las opiniones de muchosensayos hechos por voluntarios que escuchan una muestra de tráfico de voz ycalifican la calidad de la transmisión. Los voluntarios escuchan una variedad demuestras de voz, donde se les piden considerar diversos factores, como pérdidade paquetes, ruido en el circuito, eco, distorsión, retraso de paquetes y otrosproblemas de transmisión. Luego los voluntarios califican las muestras de voz, conuna calificación de 1 a 5, siendo 5 "excelente" y 1 "malo". Lasmuestras de voz son conferidas al MOS. Una puntuación de 4 en el MOS, significatener una calidad igual de buena, que en la red pública telefónica. Ver figura 3.1.

5http://www.davidwall.com/MOSCalc.htm

Figura 2-X: Escala de MOS

2.6 ASTERISK

Asterisk es una aplicación de software libre (bajo licencia GPL) que proporciona funcionalidades de una central telefónica (PBX). Como cualquier PBX, se puede conectar un número determinado de teléfonos para hacer llamadas entre sí e incluso conectar a un proveedor de VoIPo bien a una RDSI tanto básicos como primarios.

Mark Spencer, de Digium, inicialmente creó Asterisk y actualmente es su principal desarrollador, junto con otros programadoresque han contribuido a corregir errores y añadir novedades y funcionalidades.

Asterisk incluye muchas características anteriormente sólo disponibles en costosos sistemas propietarios PBX como buzón de voz, conferencias, IVR, distribución automática de llamadas, y otras muchas más. Los usuarios pueden crear nuevas funcionalidades escribiendo un dialplan en el lenguaje de

scriptde Asterisk o añadiendo módulos escritos en lenguaje C o en cualquier otro lenguaje de programación soportado por Linux.

2.6.1 Historia

Asterisk fue creada en 1999 por Mark Spencer de la empresa Digium y donada a la comunidad con licencia libre tras lo cual se han recibido muchas colaboraciones y mejoras por parte de muchos desarrolladores libres y empresas sin solicitar nada a cambio.

Poco a poco, esta aplicación se ha convertido en la evolución de las tradicionales centralitas analógicas y digitales permitiendo también integración con la tecnología más actual: VoIP. Asterisk se convierte así en el mejor, más completo, avanzado y económico sistema de comunicaciones existente en la actualidad.

Otro aliciente es su capacidad de ser programada, permitiendo realizar labores que hasta el día de hoy lo llevaban realizando sistemas extremadamente costosos y complicados y, gracias a Asterisk, esta misma labor se realiza de una forma más económica lo que fomenta el uso de sistemas libres como Linuxy estándares abiertos como SIP.

2.6.2 Estado actual

La versión estable de Asterisk está compuesta por los módulos siguientes:

Asterisk: Ficheros base del proyecto.

- DAHDI: Soporte para hardware. Drivers de tarjetas. (Anteriormente ZAPTEL)

- Addons: Complementos y añadidos del paquete Asterisk. Opcional.
- Libpri: Soporte para conexiones digitales. Opcional.
- Sounds: Aporta sonidos y frases en diferentes idiomas.

Cada módulo cuenta con una versión estable y una versión de desarrollo. La forma de identificar las versiones se realiza mediante la utilización de tres números separados por un punto. Teniendo desde el inicio como primer número el uno, el segundo número indica la versión, mientras que el tercero muestra la revisión liberada. En las revisiones se llevan a cabo correcciones, pero no se incluyen nuevas funcionalidades.

En las versiones de desarrollo el tercer valor siempre es un cero, seguido de la palabra "beta" y un número, para indicar la revisión.

2.6.3 Funcionalidades Generales

Asterisk es capaz de trabajar con prácticamente todos los estándares de telefonía tradicional:

- Lineas analógicas
- Lineas digitales: E1, T1, accesos básicos

Soporta casi todos los protocolos de VozIP:

- SIP
- IAX2
- MGCP
- Cisco Skinny

2.6.4 Esquema Conceptual

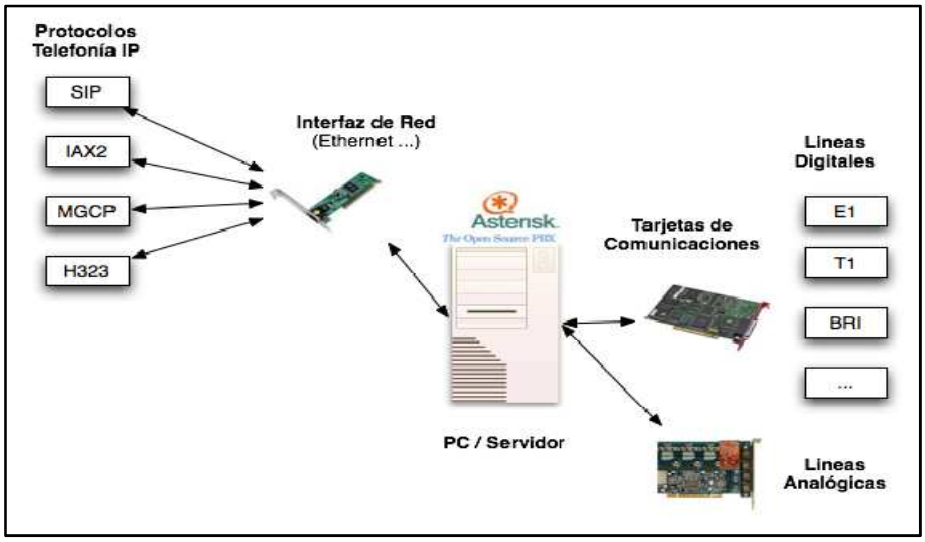


Figura 2-XI: Esquema conceptual Asterisk

2.6.5 Arquitectura Base

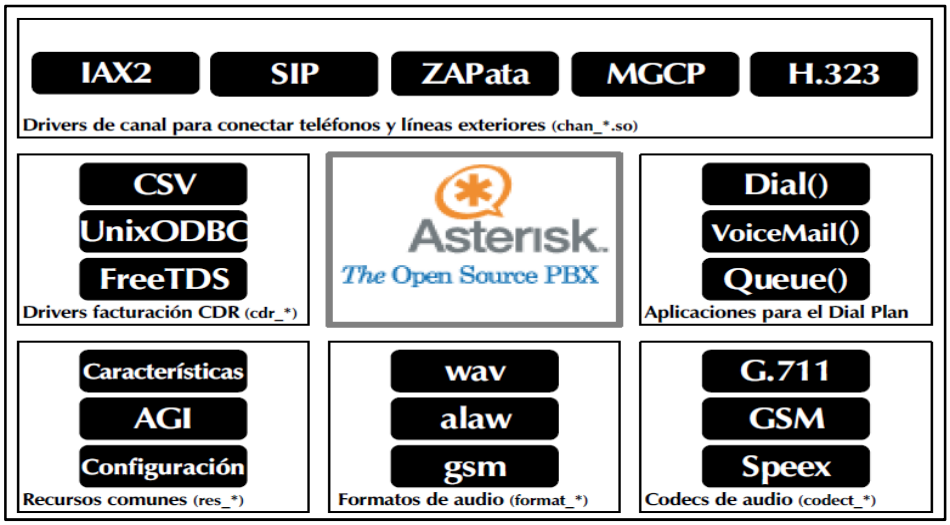


Figura 2-XII: Arquitectura Base de Asterisk

2.6.6 Funcionalidades

2.6.6.1 Tipo Centralita

Algunas de las funcionalidades, tipo centralita, más interesantes:

- Música en espera
- Registro de llamadas en BD
- Buzón de Voz por Mail
- Llamada en espera
- Salas de Conferencia
- Caller ID
- Buzón de Voz personal
- Bloqueo de Caller ID
- Colas de llamada
- Timbres distintivos
- Colas con prioridad

2.6.6.2 Funcionalidades Avanzadas

- IVR: Interactive Voice Response, gestión de llamadas con menús interactivos.
- LCR: Least Cost Routing, encaminamiento de llamadas por el proveedor VoIP más económico.
- AGI: Asterisk Gateway Interface, integración con todo tipo de aplicaciones externas.
- AMI: Asterisk Management Interface, gestión y control remoto de Asterisk.

- Configuración en base de datos: usuarios, extensiones, proveedores.

2.6.7 Requisitos Técnicos del sistema

Previa la instalación de Asterisk, es necesario contar con los requerimientos mínimos para poder ser instalado.

- Procesador a 500MHz (Pentium3) con 128 MB en RAM
- 2GB en disco duro como mínimo.
- Recomendados:
- Procesador a 1.5 GHz (Pentium 4)
- 256 MB en RAM
- 10 GB en disco duro.

2.6.8 Elección del sistema operativo

Asterisk puede ser instalado en las siguientes plataformas:

- GNU/Linux 2.x
- MacOSX 10.x
- BSD
- MS Windows

En este documento se detallará la instalación en plataformas GNU/Linux debido a que la telefonía es un servicio totalmente crítico y la elección de la plataforma donde se instalará Asterisk es clave. La estabilidad de las plataformas BSD y GNU/Linux está más que probada por infinidad de usuarios.

2.6.9 Administración

2.6.9.1 Arranque

Asterisk es un demonio que se ejecuta en segundo plano. Se invoca con el comando "*asterisk*", una vez ejecutado nos devuelve el control de la shell, haciendo un 'detach' podemos comprobar que se está ejecutando correctamente con un listado de procesos habitual:

ps aux | grep Asterisk

2.6.9.2 Conexión al CLI

En este punto tenemos el programa Asterisk en funcionamiento con la configuración de */etc/Asterisk.*

Asterisk soporta un intérprete de comandos (CLI: Command Line Interface), del estilo de muchos routers y para conectarse basta con ejecutar el comando:

asterisk r

El intérprete de comandos de Asterisk es bastante potente, y permite controlar y monitorizar gran parte de la situación de la centralita. Soporta el empleo de la tecla <Tabulador>, al estilo de las consolas de UNIX/GNU Linux, por lo que para ver un listado de todos los comandos disponibles, basta con presionar varias veces la tecla.

Para ver los posibles argumentos de un comando o completar un parámetro largo o complicado.

Como primer comando del CLI, podemos probar a verificar la versión de Asterisk instalada:

CLI> show version

Asterisk 1.6..1 built by root @ pbuxubuntu01 on a i686 running Linux

on 20060117

23:08:46 UTC

Confirmamos que Asterisk 1.6.1 se encuentra en ejecución correctamente.

Obtención del tiempo en ejecución:

CLI> show uptime

System uptime: 5 weeks, 5 days, 2 hours, 29 minutes, 28 seconds

CLI> detención

Es posible realizar una desconexión del CLI de Administración con 'quit'. Asterisk continuará ejecutándose en segundo plano.

Para matar al propio Asterisk desde el CLI, se puede utilizar el comando stop, en sus tres variantes:

- stop now: Detiene Asterisk al momento
- stop when convenient: Detiene Asterisk cuando no haya carga.
- stop gracefully: Detiene asterisk cuando no haya carga y deja de aceptar peticiones de llamadas a a partir de este momento.

2.6.9.3 Verbose

Nivel de "Verbose" es el valor que indica la cantidad de mensajes que se recibirán sobre los eventos generales del sistema. Cuanto más alto, más información sobre lo que sucede en la centralita se recibirá, este nivel, se puede establecer de varias formas:

Al arrancar el demonio:

sudo asterisk vvvvvv

Al conectarse al demonio:

sudo asterisk rvvvvvvvv

Desde el CLI:

CLI> Set Verbose 30

2.6.9.4 Debug

Nivel de "Debug" es el valor que indica la cantidad de mensajes que se recibirán sobre los eventos generales del sistema, pero utilizado normalmente para depurar problemas de drivers o de aplicaciones.

Este nivel, se puede establecer de varias formas:

Al arrancar el demonio:

sudo asterisk dddd

Al conectarse al demonio:

sudo asterisk rdddd

Desde el CLI:

CLI> Set Debug 30

2.6.10 Terminología

Canal: Es una conexión que conduce una llamada entrante o saliente en el sistema Asterisk. La conexión puede venir o salir hacia telefonía tradicional analógica o digital o VozIP.

Por defecto Asterisk soporta una serie de canales, los más importantes:

- H.323, IAX2, SIP, MGCP: Protocolos VozIP
- Console: GNU Linux OSS/ALSA sound system.
- Zap: Lineas analógicas y digitales.

Dialplan: Se trata de la configuración de la centralita Asterisk que indica el itinerario que sigue una llamada desde que entra o sale del sistema hasta que llega a su punto final.

Se trata en lineas generales del comportamiento lógico de la centralita.

Extension: En telefonía tradicional, las extensiones se asocian con teléfonos, interfaces o menús. En Asterisk, una extensión es una lista de comandos a ejecutar.

Las extensiones se acceden cuando:

- Se recibe una llamada entrante por un canal dado.
- El usuario que ha llamado marca la extensión.

- Se ejecuta un salto de extensiones desde el Dialplan de Asterisk.

Contexto (Context): El Dialplan o lógica de comportamiento de Asterisk se divide en uno o varios contextos. Un contexto es una colección de extensiones.

Los contextos existen para poder diferenciar el 'lugar' donde se encuentra una llamada, para:

- Aplicar políticas de seguridad: Asterisk no se comporta igual cuando llama un usuario y marca el 1 y cuando un usuario local marca el mismo 1.
- Menús y submenus diferenciados.
- En general, es una forma de diferenciación.

Aplicación (Application): Asterisk ejecuta secuencialmente los comandos asociados a cada extensión. Esos comandos son realmente aplicaciones que controlan el comportamiento de la llamada y del sistema en sí. Algunos ejemplos:

- Hangup: Colgar la llamada.
- Monitor: Comenzar la grabación a disco de la llamada.
- Dial: Realiza una llamada saliente.
- Goto: Salta a otra extensión o contexto.
- PlayBack: Reproduce un fichero de sonido.

2.6.11 Configuración de Asterisk

Asterisk puede configurarse desde varios puntos, los más importantes son:

- Pare desde el propio CLI
- Desde los ficheros de configuración (.conf) en /etc/asterisk

La configuración se carga al iniciar Asterisk, por lo que para aplicar cualquier cambio será necesario recargarla, para ello basta con ejecutar el comando reload en el cli:

CLI> reload

2.6.11.1 Ficheros de Configuración más importantes

Asterisk se configura desde múltiples ficheros de configuración, cada uno para una determinada área los más importantes son:

Fichero de configuración maestro: asterisk.conf

Fichero de configuración de módulos: modules.conf

Canales:

- iax.conf: Canales Inter Asterisk eXchange
- sip.conf: Canales SIP
- zapata.conf: Telefonía analógica y digital
- h323.conf: Canales H323
- mgcp.conf: Canales MGCP

Dialplan:

- extensions.conf: El propio Dialplan.
- features.conf: Dialplan para métodos complementarios

- (transferencias, call parking, grabación de llamadas bajo demanda)

Configuración de aplicaciones del Dialplan:

- meetme.conf: Para salas de conferencias.
- musiconhold.conf: Configuración de la música en espera.
- queues.conf: Configuración de Colas de llamadas.
- voicemail.conf: Configuración de los buzones de Voz.

2.6.11.1.1 Configuración para canales de Voz IP SIP

Los ficheros a manipular son sip.conf e iax.conf, la instalación crea ficheros de ejemplo con la syntaxis bastante comentada a modo de guía.

2.6.11.1.2 SIP.CONF

En este fichero se definen:

- Variables generales de SIP.
- Clientes SIP.
- Servidores SIP.

Sección General

En primer lugar existe la sección [general], donde se definen variables globales y aspectos por defecto para todos los canales SIP.

La syntaxis es la siguiente:

[general]

variable1=valor1

variable2=valor2

....

register => usuario : password @ servidorregistrar

register => ....

Register pide a Asterisk que registre su presencia en el SIP, de esta forma, el proveedor sabrá 'donde estamos', solo vale para esa localización. En ningún caso es suficiente para poder hacer llamadas.

Las variables generales más importantes son:

- allow y disallow: indican los codecs permitidos / no permitidos.
- dtmfmode: permite especificar el método por el cual seenviaran los tonos (digitos pulsados durante la conversación),

valores posibles:

- nat: Informa a Asterisk del tipo de NAT en el que se encuentra.
- externip: Dirección Pública tras el NAT.
- context: Contexto por defecto donde entraran las llamadas entrantes por SIP.
- port: Puerto en el que escuchar (5060).

Clientes y Servidores

En sip.conf se definen tanto los clientes que se conectarán a Asterisk, como los proveedores que se utilizaran para encaminar llamadas. Conceptualmente, se distinguen (versión 1.2):

- user: Envia llamadas a Asterisk
- peer: Recibe llamadas de Asterisk (proveedor).
- friend: Recibe y Envía llamadas (usuario).

La syntaxis para definir un friend o un peer es:

[nombre]

type = friend / peer

variable = valor

viarable2 = valor

Las variables más importantes que deben ser configuradas inicialmente son:

- type: peer / friend
- context: Contexto donde entraran las llamadas generadas.
- nat: Indica si el usuario o peer se encuentran tras un nat.
- host: IP remota o dynamic.
- username: nombre de usuario.
- secret: contraseña de acceso.
- allow y disallow: Configuraciones de codecs específicas para cada

friend/peer.

- qualify: Evalúa el estado del extremo SIP para conocer su accesibilidad y latencia.

2.6.12 Verificación de la configuración con el CLI

Mediante el comando "reload" en el CLI de Asterisk, le indicamos que recargue la configuración. Aunque es posible recargar de forma independiente:

CLI> sip reload

Una vez recargada, podemos comprobar los "friends" que hemos definido con el comando:

sip show users.

Para ver los "peers" definidos:

sip show peers

Es importante recalcar que los "friends" son también "peers", ya que pueden recibir y enviar llamadas.

Desde el CLI, podemos consultar si Asterisk se ha 'registrado' correctamente en los registrars configurados en la sección general con el comando:

sip show registry

2.6.13 Introducción al Dialplan

Cuando un usuario marca un determinado número la manera en la que podemos llamar utilizando alguno de los proveedores configurados es mediante el Dialplan.

El Dialplan es el corazón del comportamiento de Asterisk, en él se configura toda la lógica en lenguaje natural, un ejemplo muy sencillo podría ser el siguiente:

Cuando un usuario marca un número:

- Si el número empieza por 0, llamar al destino utilizando un proveedor externo.
- Si el número tiene 3 cifras y empieza por 1, llamar a un determinado usuario del a centralita.
- Si cuando llamamos a ese usuario, no coge en 60 segundos, reproducir un mensaje de alerta.
- En situaciones normales, el dialplan se puede complicar considerablemente..

2.6.13.1 Arquitectura del dialplan

El dialplan se define en extensions.conf, su "forma" genérica se asemeja al esquema de la Figura 2-XIII:

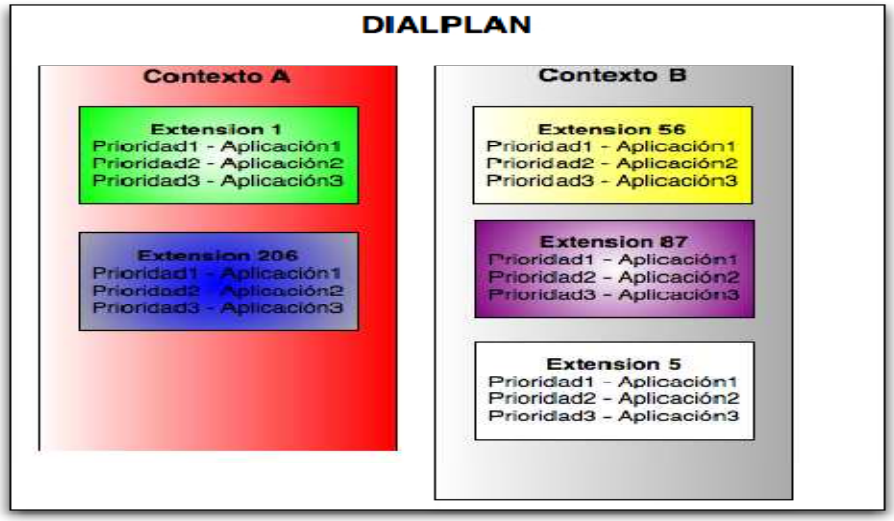


Figura 2-XIII: Dialplan

2.6.13.1.1 Aspectos a tener en cuenta

Si no existe la prioridad N + 1, Asterisk no salta a la siguiente prioridad (N+2).

Existen aplicaciones como Goto que modifican el flujo de la ejecución.

Algunas extensiones especiales:

- s: Extensión por defecto cuando una llamada entra en un contexto sin número destino asociado.
- i: Cuando el usuario marca una extensión incorrecta.
- t: Cuando se produce un timeout.

Es posible analizar cómo ha 'leido' Asterisk el fichero extensions.conf desde el CLI, con esto confirmamos posibles errores de syntaxis, etc ..

El comando es:

CLI> show dialplan [contexto]

Ejemplo:

irontur*CLI> show dialplan desde_usuarios

[Context 'desde_usuarios' created by 'pbx_config']

'_1XX' => 1. Macro(llamarusuario|${EXTEN}) [pbx_config]

Include => 'servicios' [pbx_config]

Include => 'fijos' [pbx_config]

Include => 'moviles' [pbx_config]

irontur*CLI>

=

1 extensions (1 priorities) in 1 context. =

2.6.13.2 Detalles sobre extensiones

Las extensiones son los dígitos, el destino de llamada que ha marcado el usuario cuando llama. Cuando un usuario SIP marca el 105, el flujo de ejecución salta a la extensión 105 en el contexto asociado a ese usuario SIP.

Asterisk, cuando recibe una llamada, la procesa en una determinada extensión. Pero puede quedarse a la espera ("marque el 1", "marque el 2"...) y saltar a la extensión que marca la llamada entrante: Caso de los menús IVR

2.6.13.3 Manejo de Extensiones

La syntaxis general en el dialplan es:

exten => EXTENSION, PRIORIDAD, Aplicación

En el caso de llamadas internas o funcionamiento simple, las extensiones son conocidas.

Pero cuando un usuario llama a un número que no se pre-conoce se debe utilizar patrones en las extensiones.

2.6.13.4 Patrones de Coincidencia

Para indicar patrones, se utiliza el carácter: "_"

Se pueden utilizar:

- X: Indica un dígito del 0 al 9
- Z: Indica un dígito del 1 al 9
- N: Indica un dígito del 2 al 9
- [129] Indica el 1, 2 o 9

. Indica uno o más caracteres (¡Atención! Coincide con las extensiones especiales: h,i,t .., recomendable: _X.)

Ejemplos:

- Fijos Nacionales: exten=> _9XXXXXXXX
- Internacionales: exten=> _00.

2.6.13.5 Variables

En el Dialplan de Asterisk existen variables, que pueden ser modificadas por el propio Asterisk en su ejecución lógica o por comandos expresos del Dialplan, las aplicaciones pueden cambiar variables.

Los tipos de variables son:

- Globales: Declaradas en extensions.conf (o por comando).
- Canal: Son propias a cada canal.
- Entorno: Variables de entorno (UNIX Like).

La sintaxis de una variable es:

${variable}

2.6.13.6 Manejo de Variables

Asignación de variables:

SetVar(Variable=valor)

SetGlobalVar(Variable=valor)

Manejo de cadenas:

Subcadenas: ${Variable : offset : longitud }

Devuelve la subcadena de variable que comienza en offset y con la longitud especificada.

Ejemplo:

${ 123456789:2:3} devuelve 345

Longitud:

${LEN(Variable)}

Concatenación:

${Variable1}${Variable2}

Variables de canal definidas automáticamente

Listado de variables más importantes:

- ${CALLERID}: Caller ID actual, nombre y número.
- ${CONTEXT}: Contexto actual.
- ${EXTEN}: Extensión actual.
- ${CHANNEL}: Canal actual.
- ${DIALSTATUS}: Estado de la llamada: unavailable, congestion, busy, noanswer, answer, cancel, hangup.
- ${DATETIME}: Hora actual.

Un comando útil para ver el contenido es NoOp:

NoOp (${VARIABLE})

Nos mostrará en el CLI el valor.

2.6.13.7 Expresiones

Es posible utilizar expresiones en las llamadas a aplicaciones (principalmente: GotoIf)

Syntaxis:

s[expr1 operador expr2]

Operadores Lógicos: |(or) , &(AND)

Operadores de Comparación: =, !=, <, >, <=, >=

Operadores Aritméticos: +, -, *, /, %

Ejemplos:

exten => 1,1,SetVar(total=$[1 + 1])

exten => 1,2,GotoIf($[${CALLERID}=123456]?10:20)

2.6.14 Funcionalidades

Toda la secuencia y programación del dialplan es el verdadero núcleo del sistema centralita, si bien, las siguientes funcionalidades se configuran en features.conf:

- ✓ Transferencias de llamadas: transferencia de llamadas entre diversos usuarios, independientemente de la tecnología que usen.
- ✓ Call Parking: Parking de llamadas.
- ✓ Call Pickup: Auto-transferencia de un teléfono que esté sonando.

2.6.14.1 Música en Espera

Asterisk puede poner un canal dado en espera ('HOLD'), principalmente en las siguientes situaciones:

- ✓ Durante una transferencia.
- ✓ Durante una llamada si se ha especificado el parámetro 'm', que indica que no se oirá tono de llamada sino música en espera.
- ✓ Durante una espera en el parking.
- ✓ Si la aplicación MusicOnHold o WaitMusicOnHold ha sido llamada desde el DialPlan
- ✓ Si el destino de la llamada ha solicitado explícitamente que la llamada sea puesta en espera

Es posible tener distintos tipos de música en espera. La música en espera se configura en musiconhold.conf

Asterisk puede gestionar la música en espera de varias formas:

- ✓ Utilizando mpg123: Asterisk mantiene en ejecución continua la aplicación mpg123 con un 'pipe' para el audio.
- ✓ Utilizando la música en espera en formatos nativos: Es posible tener la música en espera en formatos de audio nativos de Asterisk. Con asterisk-addons se incluye el formato: format_mp3
- ✓ Utilizando reproductores externos tipo 'madplay'.
- ✓ Utilizando mpg123:
- ✓ Solo es válido mpg123 (no mpg321), la versión recomendada es 0.59r
- ✓ Desde las fuentes de asterisk es posible descargar y compilar la versión adecuada: make mpg123

- ✓ Al arrancar Asterisk se deberán ver procesos mpg123 en ejecución contínua ('streameando' el audio al 'pipe' de Asterisk)

La configuración tipo en musiconhold.conf:

[default]

mode = quietmp3 ; (quietmp3 / mp3 / mp3nb / quietmp3nb)

directory = /var/lib/asterisk/mohmp3

El formato nativo, disponible desde asterisk 1.2.x es más estable que la combinación con mpg123

Configuración para utilizar el formato nativo:

- ✓ Es necesario compilar asterisk-addons (en concreto el directorio format_mp3)
- ✓ En /etc/asterisk/modules.conf debe indicarse la precarga del módulos: preload => format_mp3.so
- ✓ En musiconhold.conf, indicamos el tipo de música en espera que queremos (suponiendo para el modo default):

[default]

mode = files

directory = /var/lib/asterisk/mohmp3

2.6.14.2 Colas de llamadas

Una llamada entrante puede ser enviada a una cola de llamadas, que será gestionada por determinados usuarios. Se utilizan mucho en entornos tipo 'callcenter', con los canales tipo de Agentes (que hacen 'login en el sistema').

Las colas pueden comportarse de forma distinta:

- Suena todos los teléfonos hasta que alguno descuelgue.
- Los teléfonos van sonando en orden

Existen colas con prioridad. Las colas de llamadas se configuran en queues.conf:

2.6.14.3 Registro de llamadas

Asterisk permite llevar un control exhaustivo de todas las llamadas que se han realizado o recibido. Es interesante para control propio de facturación, independiente del proveedor (sino lo somos). Permite realizar estadísticas. Este control se denomina: CDR, Call Detail Record

El registro del CDR se escribe por defecto en el fichero

/var/log/asterisk/cdr-csv/Master.csv

Existen extensiones al cdr: cdr_mysql por ejemplo, que permiten almacenar los registros en una base de datos.

El CDR se configura en el fichero cdr.conf, para el módulo de MySQL, se utiliza cdr_mysql.conf

Para confirmar el estado del CDR desde el CLI, se puede ejecutar:

CLI> cdr status

Existe muchas aplicaciones que permite gestionar el CDR. Desarrollar una propia no es realmente muy complejo.

Algunas aplicaciones open source:

- Astbill: Es una de las mejores aplicaciones opensource para tarificación, control de cuentas y llamadas.
- Areski Stat v2: Se trata de una aplicación para listar y realizar estadísticas de las llamadas realizas o enviadas.
- A2Billing
- labslite: Irontec Asterisk Billing system (próximamente).

CAPÍTULO 3 III

MARCO METODOLOGICO

3.1 TIPO DE INVESTIGACION

Por la naturaleza de la investigación se considera que el tipo de estudio que se va a realizar es una investigación experimental y correlacional.

Experimental, ya que la investigación va más allá de la descripción de conceptos o fenómenos o del establecimiento de relaciones entre conceptos, sino está dirigida a responder las causas de los problemas, es decir el interés del estudio se centra en explicar cuáles son los beneficios de la telefonía ip sobre IPv6.

Correlacional, debido a que nuestro estudio contempla como se puede manipular los factores determinantes del funcionamiento de cada sistema, los cuales son el rendimiento, la escalabilidad, la facilidad de implementación y calidad de voz de la telefonía ip bajo IPV4 e IPV6, para afectar al factor óptimo funcionamiento del sistema.

Se utilizará para este proyecto los siguientes métodos de investigación:

Método Científico y de Observación: ya que se tendrá que estudiar y detectar ciertos rasgos de los protocolos propuestos para las tecnologías de VoIP

Método Inductivo: Debido que al observar particularmente el funcionamiento de los protocolos en las tecnologías VoIP, se va a llegar a una conclusión que permita identificar las diferencias y mejoras de VoIP sobre IPv6 en relación a IPv4.

Método de Análisis: Ya que para llegar a la conclusión se tendrá que desglosar todos los problemas del protocolo IPv4, y así asociar una relación causa-efecto para su comprensión.

Métodos Empírico, Experimental, Comparativo y Estadístico: Para complementar procesos que se ejecutarán dentro de la investigación.

Se ha realizado las siguientes consideraciones para esta investigación:

- ✓ Se plantea la investigación en base a los problemas existentes en la VoIP y el protocolo de red IPv4.
- ✓ Se trazan los objetivos de la investigación que determinarán los problemas de la VoIP bajo el protocolo IPv4 y los beneficios al implementarlo bajo ipv6.
- ✓ Se justifica los motivos por los cuales se propone realizar la presente investigación.
- ✓ Se elabora un marco teórico que ayude a tener una idea general para la realización del trabajo y un horizonte más amplio.

- ✓ Se plantea una hipótesis la cual es una posible repuesta al problema planteado y posee una íntima relación entre el problema y el objetivo.
- ✓ Se propone la operacionalización de las variables en base a la hipótesis planteada.
- ✓ Se define las unidades de análisis y se delimita la población que va a ser comparada en relación a la propuesta de la investigación.
- ✓ Se realiza la recolección de datos de los índices e indicadores respectivos mediante la observación directa y los tests.
- ✓ Se realiza la prueba de la hipótesis con los resultados obtenidos.
- ✓ Se elabora las conclusiones y recomendaciones producto de la investigación realizada.

3.2 SISTEMA DE HIPOTESIS

El estudio comparativo entre un sistema de VoIP y telefonía IP a través de IPv6 e IPv4 permitirá determinar las ventajas de IPv6.

3.3 OPERACIONALIZACION DE LAS VARIABLES

De acuerdo a la hipótesis planteada se han identificado las siguientes variables:

Variable Independiente:

- Análisis de los sistemas de VoIP a través de IPv4 e IPv6.

Variables Dependientes:

- Calidad
- Validación del Sistema
- Escalabilidad
- Facilidad de implementación

3.3.1 Operacionalización Conceptual

Tabla 3-I: Operacionalización Conceptual

VARIABLE	TIPO	DEFINICION
V1. Análisis de los sistemas de VoIP a través de IPv4 e IPv6.	Independiente	Estudio de las características principales de los sistemas VoIP (IPv4 e IPv6)
V2. Calidad	Dependiente	Velocidades de comunicación, intercambio de paquetes.
V3. Escalabilidad	Dependiente	Capacidad del sistema informático de cambiar su tamaño o configuración para adaptarse a las

Tabla 3-I: Operacionalización Conceptual (Continuación…)

VARIABLE	TIPO	DEFINICION
		circunstancias cambiantes.
V4. Facilidad de implementación	Dependiente	Grado de dificultad al realizar el proceso de implementación del sistema VoIP
V5. Validación del Sistema	Dependiente	Nivel de satisfacción del cliente en el manejo y funcionamiento del sistema VoIP.

Fuente: Los Autores de esta investigación

3.3.2 Operacionalización Metodológica

Tabla 3-II: Operacionalización Metodológica

VARIABLES	INDICADORES	TECNICAS	INSTRUMENTOS
V1. Independiente Análisis de los sistemas de VoIP a través de IPv4 e IPv6.	I1. Etiquetado de flujo I2. Seguridad de los datos I3. Clasificación de tráfico I4. Fragmentación	Observación Razonamiento Recopilación de información Análisis Lectura científica	✓ Iniciativas ✓ Intuición ✓ Razonamiento

V2. Dependiente Calidad	I5. MOS	Pruebas Conclusiones	✓ Iniciativas ✓ Intuición ✓ Simulaciones ✓ Razonamiento ✓ Sniffer ✓

Tabla 3-II: Operacionalización Metodológica (Continuación…)

VARIABLES	INDICADORES	TECNICAS	INSTRUMENTOS
V3. Dependiente Escalabilidad	I6. Capacidad de expansión y adaptación.	Pruebas Conclusiones	✓ Simulaciones ✓ Razonamiento
V4. Dependiente Facilidad de implementación	I7. Instalación y configuración I8. Documentación	Pruebas Conclusiones	✓ Simulaciones ✓ Razonamiento
V5. Dependiente Validación del	I9. calidad de llamada según encuesta	Pruebas Conclusiones	✓ Encuesta ✓ Simulación

Sistema	I10. facilidad de uso I11. calidad de voz en capturas		✓ Razonamiento

Fuente: Los Autores de esta Investigación

3.4 DESCRIPCIÓN DE LAS VARIABLES Y SUS RESPECTIVOS INDICADORES

Para el Análisis de los sistemas de VoIP bajo IPv4 e IPv6, se determinaron ciertos indicadores que nos servirán de base para demostrar la mejor opción para la implementación de una central de telefonía IP.

3.4.1 V1. INDEPENDIENTE: ANÁLISIS DE LOS SISTEMAS DE VOIP A TRAVÉS DE IPV6 E IPV4.

3.4.1.1 INDICADORES

I1. ETIQUETADO DE FLUJO

Combinación de la dirección de fuente y una etiqueta de flujo asignada a cada paquete de datos, con lo cual, todos los paquetes que formen parte del mismo flujo tienen asignada la misma etiqueta de flujo por parte de la fuente, lo cual permite una distribución de contenido multimedia de manera eficiente y óptima.

I2. SEGURIDAD DE LOS DATOS

Capacidad de proteger los datos ante amenazas como la escuchadisimulada y el hacking[6].

6Hacking – Delito informático que burla seguridades de una red o tecnología

I3. CLASIFICACIÓN DE TRÁFICO

Identificación y asignación de prioridades de los paquetes a ser enviados, para brindar la posibilidad de realizar un control de congestión del tráfico en la red.

I4. FRAGMENTACIÓN

Capacidad del sistema para evitar los problemas q trae la fragmentación en la red como son la sobrecarga de fragmentos en un nodo,sobrecarga deprocesamiento en los equipos de red y la pérdida defragmentos.

3.4.2 V2. DEPENDIENTE: CALIDAD

3.4.2.1 INDICADORES

3.4.2.1.1 MOS

Valor que representa una calificación subjetiva usada para medir la calidad de las conexiones telefónicas.

3.4.3 V3. DEPENDIENTE: ESCALABILIDAD

3.4.3.1 INDICADORES

3.4.3.1.1 CAPACIDAD DE EXPANSIÓN Y ADAPTACIÓN.

Capacidad del sistema informático de cambiar su tamaño o configuración para adaptarse a las circunstancias cambiantes.

3.4.4 V4. DEPENDIENTE: FACILIDAD DE IMPLEMENTACIÓN

3.4.4.1 INDICADORES

I7. INSTALACIÓN Y CONFIGURACIÓN

Proceso o pasos para poner a funcionar correctamente un sistema VoIP sobre algún protocolo.

I8. DOCUMENTACIÓN

Bajo este término se agrupan todos los manuales, guías de referencias, libros de ayuda, Internet, etc. Explicando el Qué, Cómo y el Porqué del sistema VoIP.

3.4.5 V5. DEPENDIENTE: VALIDACIÓN DEL SISTEMA

3.4.5.1 INDICADORES

I9. CALIDAD DE LLAMADA SEGÚN ENCUESTA

Apreciación de la calidad de voz experimentada en cada sistema VoIP obtenida de la Encuesta realizada a un grupo de personas.

I10. FACILIDAD DE USO

Apreciación del manejo de los sistemas VoIP obtenidos de la Encuesta realizada a un grupo de personas.

I11. CALIDAD DE VOZ EN CAPTURAS

Análisis visual y auditivo de las capturas de audio de las llamadas de prueba realizadas con Wireshark (Anexo 6 y 7).

3.5 POBLACION Y MUESTRA

La población es el conjunto de todos los elementos a ser evaluados y en la presente investigación la conforman los Clientes de sistemas de VoIP y telefonía IP concretamente aquellos que utilizan el IP PBX Asterisk.

De esta población se seleccionó una muestra no probabilística, esta es la red de pruebas implementada específicamente para investigar y realizar las pruebas de este documento junto con un grupo de personas q validaran el funcionamiento del sistema.

Esta muestra se seleccionó en base a que esta red cuenta con la infraestructura necesaria para la implementación y pruebas requeridas lo que nos permitirá realizar nuestra investigación en un ambiente real.

3.6 PROCEDIMIENTOS GENERALES

Se ha procedido a detallar los métodos utilizados en la presente investigación:

METODO: Comparativo – experimental

TECNICAS: Experimentos y pruebas

INSTRUMENTOS: Sniffer, Encuesta

3.7 INSTRUMENTOS DE RECOLECCION DE DATOS

De acuerdo a la naturaleza de la investigación, los instrumentos más apropiados para la recolección de datos fueron la comparación de

experimentos, pruebas y encuestas, los mismos que se aplicaron utilizando una serie de pruebas.

Para la recolección de información se utilizó para ciertos casos la observación directa para comparar niveles de rendimiento de la VoIP bajo IPv4 e IPv6, esto ayudándonos con la configuración del analizador de paquetes también llamado sniffer, el elegido fue Wireshark.

De la misma manera se comparó niveles de referencia de los experimentos y de las encuestas, en relación a la implementación y funcionamiento de la VoIP sobre IPv4 e IPv6 y así poder determinar los grados de fiabilidad en la comunicación en la red, utilizando también el sniffer Wireshark y la encuesta.

El estudio determina el mejor rendimiento q ofrece la implementación de un sistema de VoIP sobre IPV6.

3.8 VALIDACION DE LOS INSTRUMENTOS

La validez de los instrumentos depende del grado en que se mide el dominio específico de las variables que intervienen en la investigación. Todo instrumento aplicado debe tener como característica fundamental: la validez y la confiabilidad. La validez se refiere al grado en que un instrumento realmente mide la variable que pretende medir.

Para el análisis en capa 2 o capa de enlace de datos se utilizó como se mencionó un Analizador de Paquetes o sniffer. Estas herramientas leen el tráfico de las redes que se encuentran en su alcance y permiten almacenarlo

en ficheros para su posterior procesamiento. Se eligió la herramienta Wireshark[7], que es un analizador de protocolos utilizado para realizar análisis y solucionar problemas en redes de comunicaciones para desarrollo de software y protocolos, y como una herramienta didáctica para educación.

La principal razón de la elección de Wireshark es una de las grandes diferencias de esta aplicación con respecto a otras de este tipo, ya que permite analizar el tráfico tanto de paquetes IPv4 como de IPv6[8].

Para la evaluación del nivel de satisfacción del cliente en el manejo y funcionamiento del sistema VoIP se utilizó la encuesta[9] como herramienta para determinar los estados de opinión o hechos específicos que los usuarios experimentan en el manejo de un sistema VoIP.

7 http://www.wireshark.org/

8 http://seguridadyredes.nireblog.com/post/2010/04/05/wireshark-captura-conversaciones-voip-protocolo-sip-sdp-y-rtp-extraccion-de-audio

9 http://es.wikipedia.org/wiki/Encuesta

CAPÍTULO 4 IV

ANÁLISIS E INTERPRETACIÓN DE RESULTADOS

4.1 PROCEDIMIENTO

La forma principal para determinar las diferencias en el funcionamiento de la VoIP en las redes IPv4 e IPv6 es detectando variantes en aspectos como velocidad de respuesta, integridad de paquetes, calidad de voz. Para lo cual se realizó la evaluación del sistema en un ambiente de pruebas conformado por dos equipos Clientes y un equipo servidor de los servicios de telefonía IP.

Los métodos a utilizarse para la determinación las diferencias en el funcionamiento de la VoIP en las redes IPv4 e IPv6 incluyen al cálculo de la pérdida de paquetes, el jitter, el retraso de paquetes, la calidad de audio, el índice MOS y la evaluación mediante una encuesta realizada a una muestra de personas que utilizaron el sistema de VoIP implementado en IPv4 e IPv6.

4.2 PROCESAMIENTO DE LA INFORMACION

Se realizó un análisis tomando en cuenta cada uno de los factores de las variables independientes y evaluando a cada factor según los valores recomendados en las diferentes bibliografías consultadas. Todos los datos obtenidos ya sean cuantitativos o cualitativos se convierten a valores cualitativos en un mismo rango general.

A cada variable independiente se le asignó pesos dados por su importancia e influencia a la variable dependiente, con estos pesos se determinó un peso matemático.

$$ValorPonderado = PesoMatemático \times ValorCualitativo$$

Con los valores cualitativos y el peso matemático de cada variable se obtuvo una medida ponderada aplicando la **¡Error! No se encuentra el origen de la referencia.**.

Para obtener calificación total de cada protocolo se utilizó la Ecuación 0.a

$$\sum_{i=0}^{n} ValorPonderado_i$$

Ecuación 0.a: Sumatoria de Valores Ponderados

4.3 RESUMEN DE LOS EXPERIMENTOS DE EVALUACIÓN DE FUNCIONAMIENTO

Para el análisis de las diferencias en el funcionamiento de la VoIP en las redes IPv4 e IPv6 se realizaron varios experimentos con diferentes condiciones y parámetros, así como la evaluación mediante una encuesta realizada a una muestra de personas que utilizaron el sistema de VoIP implementado en IPv4 e

IPv6. Estas pruebas que se exponen en los anexos de esta tesis ayudarán a entender el funcionamiento de estos dos sistemas o bien, entender los beneficios existentes al implementar un sistema de VoIP conjuntamente con el protocolo IPv6.

Para cada sistema VoIP, se consideraron tráfico IP al realizar llamadas entre los clientes, los parámetros solicitados son tiempo e integridad de paquetes. En cuanto al tiempo, se refiere al intervalo de respuesta obtenida en todos los indicadores presentes en el proceso de llamada.

Para la evaluación se realizó una encuesta para determinar parámetros como la calidad de voz, tiempo de respuesta, facilidad de manejo experimentada en cada sistema VoIP

4.3.1 Ambiente de Simulación

La Figura 4.1 muestra el ambiente de simulación experimental compuesta por dos clientes y un servidor de servicios VoIP.

Este ambiente fue configurado como una red LAN, la implementación consiste en tres equipos conectados a la LAN, dos Netbooks que son los clientes dotados de Softphones y una computadora de escritorio la misma q aloja al servidor de servicios VoIP

En el escenario de la

Figura 4-I se configuró un Analizador de paquetes como Wireshark[10], las características y configuración de este programa se muestran en el Anexo 5.

10 Wireshark es un programa para Linux que permite detectar y analizar paquetes en redes IPv6 e IPv4

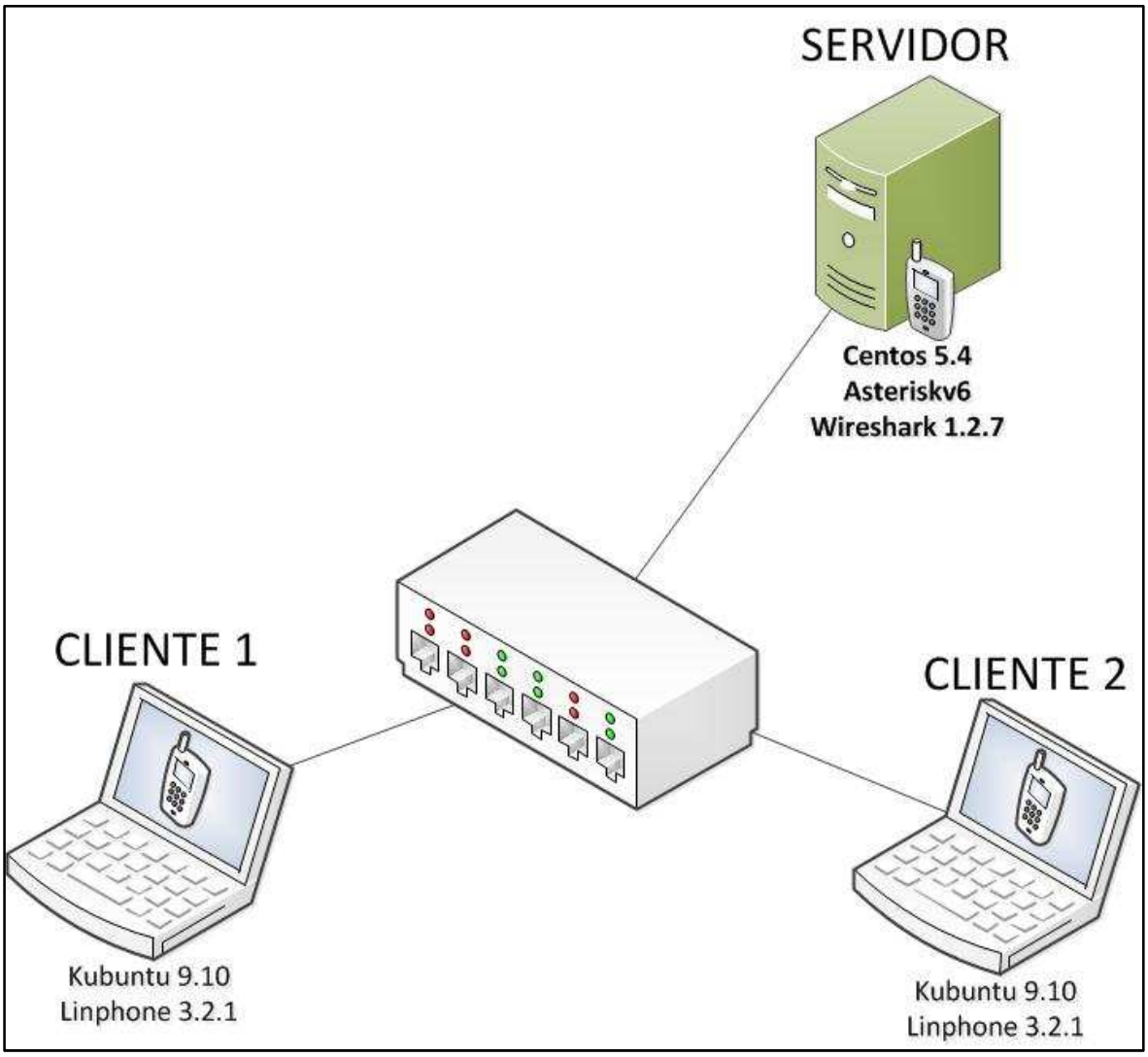


Figura 4-I: Ambiente de Simulación Experimental

Tabla 4-I: Detalles Técnicos de los equipos del Ambiente de Simulación

Cantidad	Equipo	Descripción
1	Switch	SwitchD-Link DIR-600 de 8 puertos
1	Estación portátil	Marca: Acer Aspire One Sistema Operativo: Kubuntu 9.10

		Memoria: 1GB Procesador: Intel Atom 1.6Ghz
1	Estación portátil	Marca: Xtratech Sistema Operativo: Kubuntu 9.10 Memoria: 2GB Procesador: Intel Atom 1.6Ghz
1	Servidor Asterisk	Sistema Operativo: Centos 5.4 Memoria: 1GB Procesador: Intel Pentium IV 3.0GHz

Fuente: Los Autores de esta investigación

4.3.2 ESTUDIO COMPARATIVO DE LA VARIABLE INDEPENDIENTE

La escala de valoración cualitativa para los indicadores de la variable independiente a ser utilizada para obtener el total será:

Tabla 4-II: Cuantificadores de calificación de los parámetros de V1

1	2	3
Ineficiente.	Poco eficiente.	Eficiente y óptimo

Fuente: Los Autores de esta investigación

Tabla 4-II: Comparativa de VoIP IPv4 y VoIP IPv6

INDICADORES	VOIP IPV4	VOIP IPV6
I1. Etiquetado de flujo	1	3
I2. Seguridad de los datos	1	3
I3. Clasificación de tráfico	2	3
I4. Fragmentación	2	3
TOTAL V1	**6**	**9**

Fuente: Los Autores de esta investigación

El cuadro anterior muestra las características principales que VoIP IPv6tiene frente a VoIP IPv4 y justifica el porqué es mejor laimplementación de VoIP sobre IPv6, los datos de la tabla anterior están expresados en forma gráfica en la figura 4.2 donde puede evidenciar claramente como la implementación de VoIP sobre IPv6 es superior a la implementación de VoIP que está sobre IPv4.

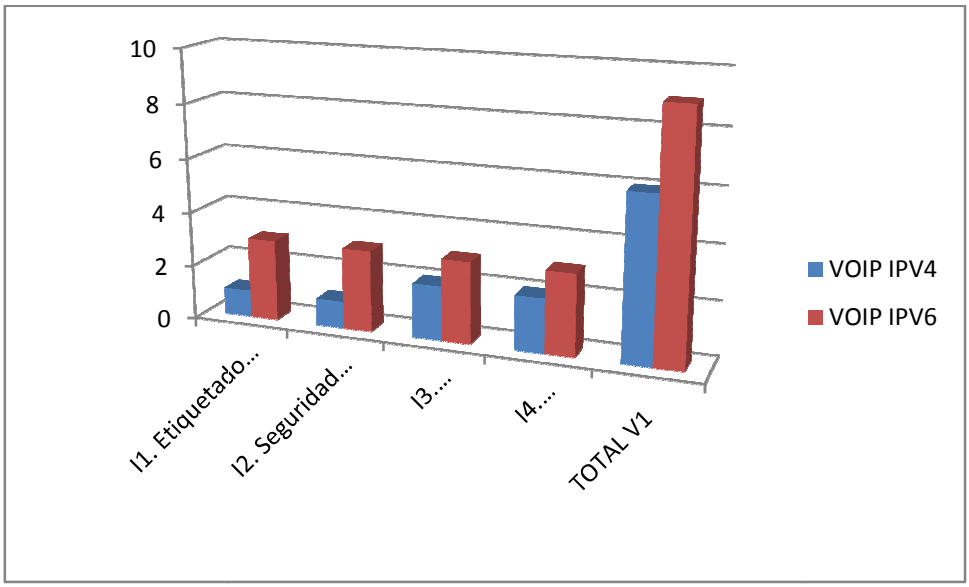


Figura 4-II: VoIP IPv4 vs. VoIP IPv6

4.3.3 ESTUDIO COMPARATIVO DE LAS VARIABLES DEPENDIENTES

Para demostrar las variables dependientes y continuar con el análisis e interpretación de resultados, se procedió con la recopilación de datos de cada sistema los cuales serán analizados y procesados.

4.3.3.1 V2: CALIDAD VARIABLE DEPENDIENTE

4.3.3.1.1 INDICADOR 5: MOS

Para medir la Calidad del Sistema en cada protocolo se tomó los datos obtenidos en los dos experimentos (Anexo 6 y Anexo 7) y se determinaron los valores a tomar en cuenta con la ayuda de Wireshark:

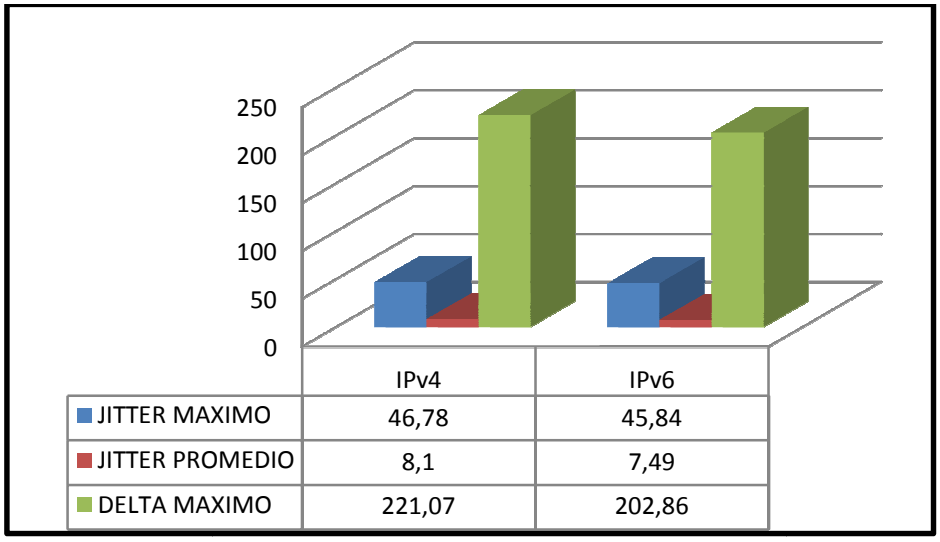
- Pérdida de Paquetes
- Jitter Máximo
- Jitter Promedio

- Retraso(Delta)

La mejor forma de relacionar estos indicadores es el índice MOS (Página 74), este índice es una medida cualitativa de la calidad de voz en una conexión telefónica.

4.3.3.1.2 Cálculos V2

La Figura 4-III muestra los valores cuantitativos obtenidos en cada protocolo mostrando una leve diferencia en cada uno de los indicadores.



	IPv4	IPv6
JITTER MAXIMO	46,78	45,84
JITTER PROMEDIO	8,1	7,49
DELTA MAXIMO	221,07	202,86

Figura 4-III: Datos obtenidos en los experimentos

La pérdida de paquetes no se muestra en la gráfica porque en ninguno de los dos experimentos se obtuvo valor alguno en dicho factor.

Con los valores de cada factor se procedió a calcular el valor del indicador MOS de cada protocolo con los datos obtenidos en los experimentos, para esto se utilizó un formulario Online que se encuentra en la siguiente dirección:

http://www.davidwall.com/MOSCalc.htm.

En las

Figura 4-IV y Figura 4-V se muestran los resultados obtenidos

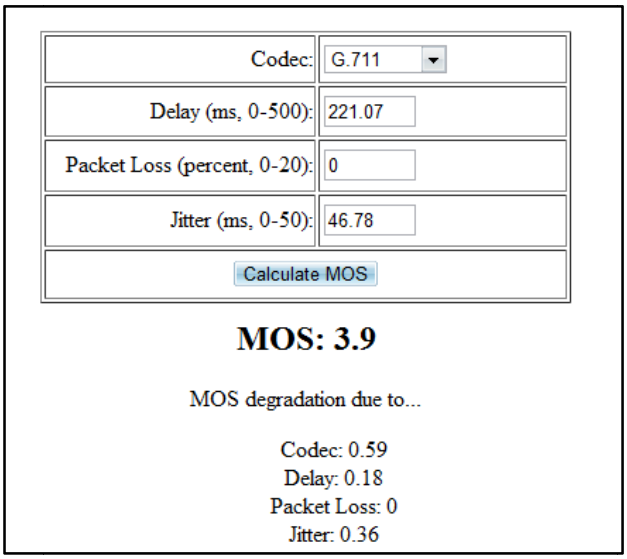


Figura 4-IV MOS IPv4

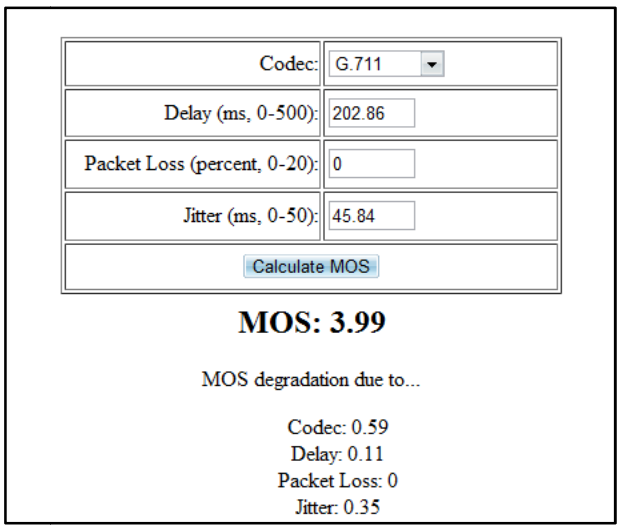


Figura 4-V: MOS IPv6

Utilizaremos la escala de valores del indicador MOS presentada a continuación para determinar el valor de cada sistema.

Figura 4-VI: Escala de valores del indicador MOS

Ahora hacemos una regla de tres simple con un valor de MOS óptimo de 5, a fin de obtener una calificación sobre 100 de cada protocolo en esta variable (Calidad):

Calculamos primero la variable Calidad en IPv4:

$$5 \quad \gg \quad 100$$

$$3.9 \quad \gg \quad Calidad\ IPv4$$

$$\text{Calidad VoIPv4} = \frac{3.9 \times 100}{5}$$

$$\boldsymbol{Calidad\ VoIPv4 = 78}$$

Ahora Calculamos la variable Calidad en IPv6:

$$5 \quad \gg \quad 100$$

$$3.99 \quad \gg \quad Calidad\ IPv6$$

$$Calidad\ VoIPv6 = \frac{3.99 \times 100}{5}$$

$$\boldsymbol{Calidad\ VoIPv6 = 79.8}$$

4.3.3.2 V3: ESCALABILIDAD VARIABLE DEPENDIENTE

4.3.3.2.1 INDICADOR 6: CAPACIDAD DE EXPANSIÓN Y ADAPTACIÓN.

El factor que realmente define la Escalabilidad es el direccionamiento de cada protocolo, en este factor se marca una gran diferencia entre los dos protocolos, IPv6 a parte de la gran cantidad de direcciones que puede manejar es un protocolo flexible que puede crecer y adaptarse a cambios en la red sin mayores dificultades.

El número de individuos conectados a Internet crece en 77 millones por año y se estima que para el 2011 los usuarios de VoIP alcancen los 250 millones[11]y si este crecimiento se mantiene la única salida para una comunicación global y duradera sería la implementación del protocolo IPv6.

Para establecer valores cualitativos diferentes para cada protocolo vamos a establecer el número de direcciones que soporta cada uno, el mayor convertirlo en el valor de referencia y mediante una regla de tres establecer la calificación del menor valor

4.3.3.2.2 Cálculos V3

Tabla 4-III: Direcciones soportadas por cada Protocolo

IPv4	IPv6
2^{32}=4294'967.296	2^{128}=340.282.366.920.938.463.463.374.607.431.768.211.456

11 http://www.itu.int/itunews/manager/display.asp?lang=es&year=2007&issue=02&ipage=futureVoice2&ext=html

Fuente: Los Autores de esta investigación

En la **¡Error! No se encuentra el origen de la referencia.**se muestran las direcciones soportadas por cada protocolo

Ya que la diferencia de los valores obtenidos es extremadamente grande no se va a calcular exactamente su valor, para cuantificar esta variable se asignará un puntaje de acuerdo a la calificación por parte de los autores utilizando la siguiente escala de valoración.

Tabla 4-IV: Cuantificadores y Abreviaturas de calificación de los parámetros de V3

0-20		**21-40**		**41-60**		**61-80**		**81-100**	
Deficiente	DF	Poco Eficiente	PE	Limitada	L	Eficiente	E	Muy Eficiente	ME

Fuente: Los Autores de esta investigación

Tabla 4-V: Calificación de V3 (Escalabilidad)

	VoIPv6		**VoIPv4**	
Calificación I5	ME	99	DF	5

Fuente: Los Autores de esta investigación

4.3.3.3 V4: FACILIDAD DE IMPLEMENTACIÓN VARIABLE DEPENDIENTE

La Implementación de los Sistemas de VoIP tanto en IPv4 como en IPv6 no presentaron grandes diferencias como se suponía al iniciar esta investigación, la mayor dificultad que se presentó fue la poca información que se pudo encontrar en cuanto al protocolo IPv6 se refiere y la compatibilidad de los

softphones con los que se experimentó. La Tabla VII muestra una lista del softphones con los cuales se experimentó y su compatibilidad con cada protocolo.

Tabla 4-VI: Compatibilidad de Softphones

Softphone	**IPv4**	**IPv6**
SJPhone	Si	No
X-Lite	Si	No
Kphone	Si	No
Linphone	Si	Si

Fuente: Los Autores de esta investigación

En cuanto a la configuración de Asterisk es independiente del protocolo, es decir los mismos archivos de configuración sirven para los dos protocolos.

A los indicadores de esta variable se les asignará un puntaje de acuerdo a la calificación por parte de los autores utilizando la siguiente escala de valoración cualitativa, los mismos que nos permitirán cuantificar las variables.

Tabla 4-VII: Cuantificadores y Abreviaturas de calificación de los parámetros de V4

0		**1**		**2**		**3**		**4**	
Difícil	D	Parcialmente Difícil	PD	Mediamente Fácil	MF	Fácil	F	Totalmente Fácil	TF
No	N							Si	S

Fuente: Los Autores de esta investigación

INDICADOR 7: INSTALACIÓN Y CONFIGURACIÓN

Tabla 4-VIII: Instalación y Configuración

	VoIPv4		VoIPv6	
Instalación de paquetes con Yum	S	4	N	0
Ausencia de errores después de la instalación	S	4	S	4
Grado de dificultad para configurar	MF	2	MF	2
Valoración Total de I6.		10		6
Porcentaje Equivalente de I6 .		83.3		50

Fuente: Los Autores de esta investigación

INDICADOR 8: DOCUMENTACIÓN

Tabla 4-IX: Documentación

	VoIPv4		VoIPv6	
Libros	TF	4	PD	1
Sitios Web	TF	4	MF	2
Valoración Total de I7.		8		3
Porcentaje Equivalente de I7.		100		37.5

Fuente: Los Autores de esta investigación

Tomando en cuenta estos hechos determinamos que la implementación del sistema en IPv6 presenta un poco más de dificultad debido a que la mayoría de aplicaciones no tienen soporte y la poca información que existe para la implementación de servicios en este protocolo.

4.3.3.3.1 Cálculos V4

Tabla 4-X: Pesos de los Indicadores de V4

Factor	Importancia	Peso Matemático
I6. Instalación y Configuración	80	0.47
I7. Documentación	90	0.53
Total	170	1

Fuente: Los Autores de esta investigación

Tabla 4-XI: Calificación de V4 (Facilidad de Implementación)

PROTOCOLOS		**VoIPv4**		**VoIPv6**	
INDICADORES	**Peso Matemático**	**Calificación**	**Ponderación**	**Calificación**	**Ponderación**
I7 Instalación y Configuración	0.47	83.3	39.15	50	23.5
I8. Documentación	0.53	100	53	37.5	19.88
TOTALES V4.			92.15		43.38

4.3.3.4 V5: VALIDACIÓN DEL SISTEMA VARIABLE DEPENDIENTE

Para la validación del sistema se realizó una encuesta a una pequeña muestra de usuarios que utilizaron el sistema de pruebas con los dos protocolos, a cada usuario se le dio una información general del sistema sin informarle las diferencias entre ellos, solo etiquetándolos como Sistema 1 (IPv4) y Sistema 2 (IPv6). La encuesta se la realizó de forma anónima para facilitar la aceptación de la misma.

También se hizo un análisis de la calidad de Audio a través de las capturas de audio de Wireshark

4.3.3.4.1 Resultados de la Encuesta (Anexo 8)

Figura 4-VII: Resultados Pregunta 1 de Encuesta

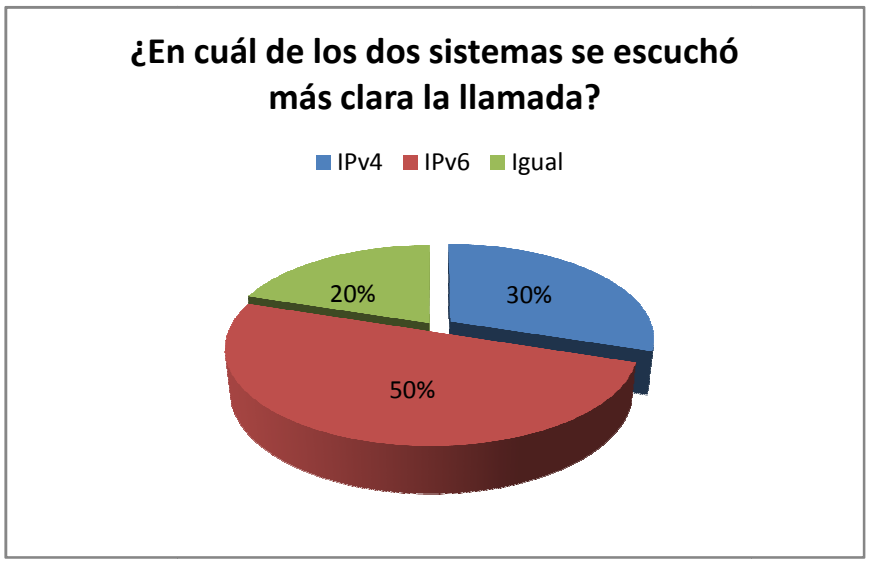


Figura 4-VIII: Resultados Pregunta 2 de Encuesta

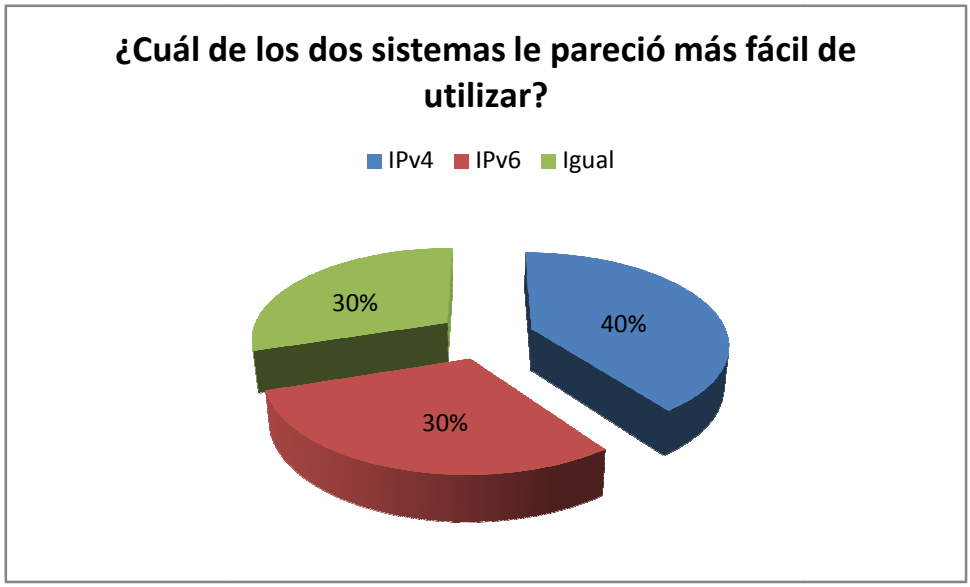


Figura 4-IX: Resultados Pregunta 3 de Encuesta

En las Figuras Figura 4-VII,Figura 4-VIII y Figura 4-IXse muestran los porcentajes de respuestas obtenidas en cada una de las preguntas de la encuesta. Las respuestas más relevantes para este análisis son las de las preguntas 2 y 3 las cuales nos indican la calidad de voz y la facilidad de uso respectivamente.

En la pregunta 2 el 50% de los encuestados opina que en el protocolo IPv6 la voz se escucha mejor que en IPv4, mientras que un 30% opinan lo contrario.

En la pregunta 3 no se obtienen mayores diferencias, sin embargo hay una pequeña ventaja del protocolo IPv4 en cuanto a la facilidad de uso, esto se debe a que algunos usuarios tuvieron problemas al acostumbrarse al nuevo formato de dirección de IPv6.

También se realizó un análisis visual y auditivo de las capturas de audio de las llamadas de prueba realizadas con Wireshark (Anexo 6 y 7) encontrándose una pequeña diferencia en el eco producido a favor del protocolo IPv6.

4.3.3.4.2 Cálculos V5

Tabla 4-XII: Pesos de los Indicadores de V5

Factor	Importancia	Peso Matemático
I8. Calidad de Llamada según Encuesta	90	0.43
I9. Facilidad de uso	90	0.43
I10. Calidad de voz en capturas	30	0.14
Total	**210**	**1**

Fuente: Los Autores de esta investigación

Tabla 4-XIII: Valores Ponderados de los Indicadores de V5

ROTOCOLOS INDICADORES	Peso Matemático	VoIPv4		VoIPv6	
		Calificación	Ponderación	Calificación	Ponderación
I9. CALIDAD DE LLAMADA SEGÚN ENCUESTA	0.43	30	12.9	50	21.5
I10. FACILIDAD DE USO	0.43	40	17.2	30	12.9
I11. CALIDAD DE VOZ EN CAPTURAS	0.14	60	8.4	80	11.2
TOTALES V5.			38.5		45.6

Fuente: Los Autores de esta investigación

4.3.4 Calificación General de Protocolos

Ahora que ya tenemos valores cualitativos en cada factor procedemos a definir sus pesos en la Tabla 4-XIV

Tabla 4-XIVPesos de cada variable

Factor	Importancia	Peso Matemático
V2. Calidad	70	0.30
V3. Escalabilidad	50	0.22
V4. Facilidad de Implementación	20	0.09
V5. Validación del Sistema	90	0.39
Total	**230**	**1**

Fuente: Los Autores de esta investigación

Con los pesos definidos calculamos los valores ponderados y la calificación Total presentados en la Tabla 4-IX.

Tabla 4-XV: Calificación General de las Variables Dependientes

Protocolos		**VoIPv4**		**VoIPv6**	
Variables Dependientes	**Peso Matemático**	**Calificación**	**Ponderación**	**Calificación**	**Ponderación**
V2. CALIDAD	0.30	78	23.4	79.8	23.94
V3. ESCALABILIDAD	0.22	5	1.1	99	21.78
V4. FACILIDAD DE IMPLEMENTACIÓN	0.09	92.15	8,29	43.38	3,9
V5. VALIDACIÓN DEL SISTEMA	0.39	38.5	15.02	45.6	17.78
TOTAL SISTEMAS			47,81		67.4

Fuente: Los Autores de esta investigación

Con lo obtenido anteriormente se puede apreciar que en general, la implementación del sistema VoIP juntamente con el protocolo IPv6 es superior a su implementación bajo IPv4.

4.4 COMPROBACIÓN DE LA HIPÓTESIS

Para la comprobación de la hipótesis planteada en la investigación debemos calcular el estadístico Chi Cuadrado a partir de los datos que se han obtenido de los resultados que se lograron del estudio comparativo de Voip y Telefonía IP en ipv6 e ipv4 a manera de cuadros comparativos, en los cuales se calificaron los indicadores de cada variable cualitativamente y cuantitativamente según el criterio de los autores basándose en los resultados teóricos y prácticos. A continuación se consideró la hipótesis nula Ho y la hipótesis de investigación Hi.

Hi: El estudio comparativo entre un sistema de VoIP y telefonía IP a través de IPv6 e IPv4 permitirá determinar las ventajas de IPv6.

Ho: El estudio comparativo entre un sistema de VoIP y telefonía IP a través de IPv6 e IPv4 no permitirá determinar las ventajas de IPv6.

Para la comprobación de la Hipótesis de la Investigación seguiremos los siguientes pasos:

4.4.1 Frecuencias Observadas

Las frecuencias observadas las obtenemos de los valores ponderados de cada variable y de cada protocolo en la tabla de calificación general de los sistemas VoIP, obteniendo la siguiente tabla. (Ver tabla 4-XI).

Tabla 4-XVI: **Frecuencias observadas**

	Ipv4	**Ipv6**	**Sumatoria de cada variable**
CALIDAD	23,4	23,94	47,34
ESCALABILIDAD	1,1	21,78	22,88
FACILIDAD DE IMPLEMENTACIÓN	8,29	3,9	12,19
VALIDACIÓN DEL SISTEMA	15,02	17,78	32,8
TOTALES	**47,81**	**67,4**	**115,21**

Fuente: Los Autores de esta investigación

La tabla XVII nos muestra la tabla de contingencia creada para el cálculo del chi cuadrado, contiene las variables estudiadas: Calidad, Validación del Sistema,Escalabilidad y Facilidad de Implementación para cada protocolo IP.

4.4.2 Frecuencias Esperadas

Las frecuencias esperadas de cada celda, se calcula mediante la siguiente fórmula aplicada a la tabla de frecuencias observadas

$$fe = \frac{(total\_de\_fila)(total\_de\_columna)}{N}$$

Ecuación 1: Fórmula para calcular la frecuencia esperada

Donde **N** es el número total de frecuencias observadas

A continuación se presentan los valores obtenidos aplicando la fórmula descrita anteriormente: (Ver tabla 4-XVIII).

Tabla 4-XVII: Frecuencias esperadas

	IPV4	**IPV6**	**Sumatoria de cada variable**
CALIDAD	19,645217	27,69478344	47,34
ESCALABILIDAD	9,494773	13,38522698	22,88
FACILIDAD DE IMPLEMENTACIÓN	5,0586225	7,131377485	12,19
VALIDACIÓN DEL SISTEMA	13,611388	19,1886121	32,8
Total	47,81	67,4	115,21

Fuente: Los Autores de esta investigación

4.4.3 Sumatoria de $\mathbf{X^2}$

Una vez obtenidas las frecuencias esperadas, se aplica la siguiente fórmula de chi cuadrado para cada una de las celdas de la tabla:

$$\mathbf{X^2} = \sum \frac{(\mathbf{O} - \mathbf{E})^2}{\mathbf{E}}$$

Ecuación 2: Fórmula para calcular ji cuadrado

Donde:

O es la frecuencia observada en cada celda

E es la frecuencia esperada en cada celda

Tabla 4-XVIII: Calculo de Chi Cuadrado

Observado(O)	Esperado(E)	(O-E)	(O-E)^2	{(O-E)^2/E}
23,4	19,6452	3,7548	14,0984	0,7177
23,94	27,6948	-3,7548	14,0984	0,5091
1,1	9,4948	-8,3948	70,4722	7,4222
21,78	13,3852	8,3948	70,4722	5,2649
8,29	5,0586	3,2314	10,4418	2,0642
3,9	7,1314	-3,2314	10,4418	1,4642
15,02	13,6114	1,4086	1,9842	0,1458
17,78	19,1886	-1,4086	1,9842	0,1034
				X^2=17,6914

Fuente: Los Autores de esta investigación

La tabla 4-XIX nos proporciona el valor X^2, para saber si ese valor es o no significativo, se debe determinar los grados de libertad mediante la siguiente fórmula.

$$\mathbf{Gl = (f - 1)(c - 1)}$$

Ecuación 3: Fórmula para calcular los grados de libertad

Donde:

f: es el número de filas de la tabla de contingencia

c: es el número de columnas de la tabla de contingencia

Gl = (4-1) (2-1) → Gl = 3

De la tabla de distribución de $\mathbf{X^2}$ que se encuentra en el Anexo 9, eligiendo como nivel de significación**: a = 0.05** con una cola **G.l = 4**, el valor crítico de la prueba $\mathbf{X_\alpha^2}$ **= 7.81**

4.4.4 Criterio de decisión

- **SI**X^2 calculado es mayor a X_α^2 (Valor crítico) de la tabla de distribución se rechaza la hipótesis nula Ho y por lo tanto se acepta la hipótesis de Investigación.
- **SI**X^2 calculado es menor a X_α^2 (Valor crítico) de la tabla de distribución se acepta la hipótesis nula Ho y por lo tanto se rechaza la hipótesis de Investigación.

4.4.5 Grafica X^2 e Interpretación

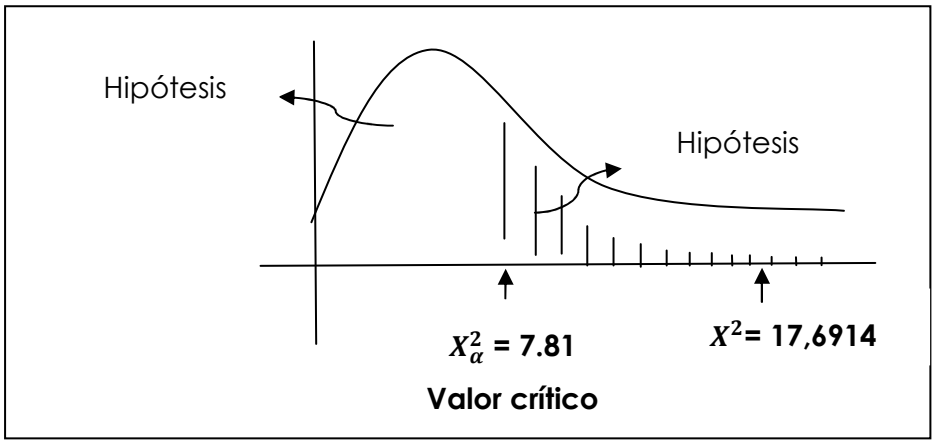


Figura 4-X: Curva del análisis del Chi Cuadrado

Como podemos observar en la Figura 4.16 el valor del estadístico Chi Cuadrado calculado (X^2 =17,6914) es mucho mayor que el nivel crítico (X_α^2 = 7.81) es decir se rechaza la hipótesis nula, por lo tanto en este caso se corrobora la hipótesis planteada en la investigación, es decir, la mejor alternativa si se puede determinar a través de un estudio comparativo entre un sistema de VoIP y telefonía IP a través de IPv4 e IPv6

CONCLUSIONES

- Se implementó un prototipo de central telefónica en IPv4 e IPv6, con los cuales se pudo comprobar el grado de desarrollo del protocolo IPv6 y su compatibilidad con el hardware y software actual.
- En cuanto a la Calidad del sistema determinado por el índice MOS obtenido de cada protocolo de acuerdo a sus valores de jitter y delta se obtuvo un 1.8% de superioridad del sistema IPv6.
- Se estableció una diferencia del 94% para el sistema bajo IPv6 en cuanto a Escalabilidad debido a la evidente superioridad del mismo en este campo.
- La facilidad de implementación experimentada en cada sistema VoIP permitió establecer una notable superioridad del sistema bajo IPv4 asignando un 47.9% de diferencia en relación al sistema bajo IPv6.
- La Validación del sistema determinado por la Calidad de Voz en llamadas, la facilidad de uso y Calidad de voz de las capturas nos dio como resultado una diferencia de 7.1% a favor del sistema IPv6.
- En la instalación y configuración del Software IP PBX Asterisk existió problemas respecto a la poca información en cuanto a su funcionamiento con el protocolo ipv6.
- Luego de varias pruebas con diferentes softphones se determinó que linphone cuenta con todas las características necesarias para la implementación del sistema de telefonía IP en cada protocolo.
- Los resultados obtenidos mediante la encuesta determinó que hay dificultad en el manejo del sistema en un entorno IPv6, pero en cuanto a

calidad de voz existió una diferencia favorable respecto al mismo.

- Mediante los resultados obtenidos en los experimentos, pruebas y análisis de éstos, se pudo apoyar la hipótesis planteada mostrando en general la superioridad de la implementación de un sistema de VoIP sobre IPV6.

RECOMENDACIONES

- La comunicación de voz a través de la red IP, de manera económica y efectiva, es un hecho, por lo que se recomienda que las empresas utilicen las posibilidades que les ofrece la red IPv6 para incrementar la productividad y competitividad.
- Es necesario hacer una revisión exhaustiva de las alternativas al momento de actualizar o implementar una red IPv6. Se descubrió que muchos fabricantes anuncian soporte IPv6 en sus productos, pero en la realidad dicho soporte es parcial o se incluirá en futuras actualizaciones. En dichos casos son útiles las iniciativas como el programa 12"IPv6 Ready" que certifican el soporte IPv6 de equipos y software, realizando una serie de pruebas sobre ellos.
- Que dentro del alcance de las organizaciones con responsabilidad en comunicaciones en cada país y específicamente en Ecuador, sean formados grupos de expertos como "Grupos de trabajo IPv6", para seguir avanzando dentro de este campo tan importante y necesario.
- Que los estados, junto con el sector privado y el académico, lleven adelante actividades para promover IPv6 en sus respectivos países.
- El uso de Wireshark es recomendado para todo Estudiante, Administrador, Investigador o Profesional en el campo de las redes de datos, ya que es una Herramienta que nos ofrece infinitas posibilidades de Análisis y Experimentación para los distintos Protocolos de Comunicación existentes

12 http://www.ipv6ready.org/

en la actualidad, siendo además Multiplataforma, Libre y con gran cantidad de Información disponible.

- El Software Libre se ha convertido hoy en día en una alternativa eficaz, robusta y accesible principalmente para investigadores que no simplemente ahorran costos sino que obtienen libertad en la elección de Herramientas, Independencia Tecnológica, Soporte de miles de usuarios y Actualizaciones gratuitas, Seguridad y sobretodo Innovación, por tanto como usuarios de varias herramientas libres utilizadas en esta investigación, no podemos dejar de recomendarlo a todo tipo de usuarios.

BIBLIOGRAFÍA

ESCUDERO, Alberto; BERTHILSON, Louise. *VOIP para el Des*arrollo. Madrid España, Ananda,2006. 300p.

LOSHIN, Pete. IPV6; Theory, Protocol and Practice. Londres Inglaterra, Cactus Press, 2004. pp 125 - 200

MINOLI, Daniel. Voice over IPv6. Burlington Estados Unidos de América, Elseiver, 2006. 381p.

MEGGELEN, Jim; SMITH, Jared. ASTERISK, the Future of Telephony. Nueva York Estados unidos de Amárica, O'Reilly, 2009. 376p.

SITIOS WEB

ASTERISK

- ✓ Sitio Oficial Asterisk: http://www.asterisk.org/

  (2010-06-25)

- ✓ Asteriskv6: http://www.asteriskv6.org/

  (2009-07-05)

- ✓ Asterisk Guru: http://www.asteriskguru.com

  (2009-07-14)

IPv6

- ✓ Sitio Oficial IPv6: http://www.ipv6.org/

  (2010-04-22)

- ✓ Wiki IPv6: http://en.wikipedia.org/wiki/IPv6

  (2010-05-04)

VoIP

- ✓ VoIP Novatos: http://www.voipnovatos.es

  (2009-10-14)
- ✓ VoIP-info: http://www.voip-info.org

  (2009-11-20)
- ✓ VoIPforo: http://www.voipforo.com

  (2010-12-13)
- ✓ VoztoVoice:http://www.voztovoice.org

  (2010-01-28)
- ✓ Linphone: http://www.linphone.org

  (2010-02-03)
- ✓ Wireshark: http://www.wireshark.org

  (2010-02-17)
- ✓ Calculo MOS: http://www.davidwall.com/MOSCalc.htm

  (2010-04-16)

ANEXOS

ANEXO 1 | INSTALACIÓN DE CENTOS 5.4

Inserte el disco DVD de instalación de [1]CentOS 5 y en cuanto aparezca el diálogo de inicio (Figura 1.1), pulse la tecla **ENTER** o bien ingrese las opciones de instalación deseadas.

Figura 1.1: Diálogo de Inicio de Centos

En la siguiente ventana (Figura 1.2) el sistema preguntará si desea verificar la integridad del disco a partir del cual se realizará la instalación, seleccione **«OK»** y pulse la tecla **ENTER**, considere que esto puede demorar varios minutos. Si está seguro de que el disco o discos a partir de los cuales se realizará la instalación están en buen estado, seleccione **«Skip»** y pulse la tecla **ENTER.**

En la ventana de Selección de Idioma (Figura 1.3) Seleccione **«Spanish»**y de click en **«Next»**

[1] **Community Enterprise Operative System**

En la ventana de la Figura 1.4 seleccione el teclado que corresponda y haga click sobre el botón **«Siguiente»**.

Figura 1.2: Ventana de Verificación de medios

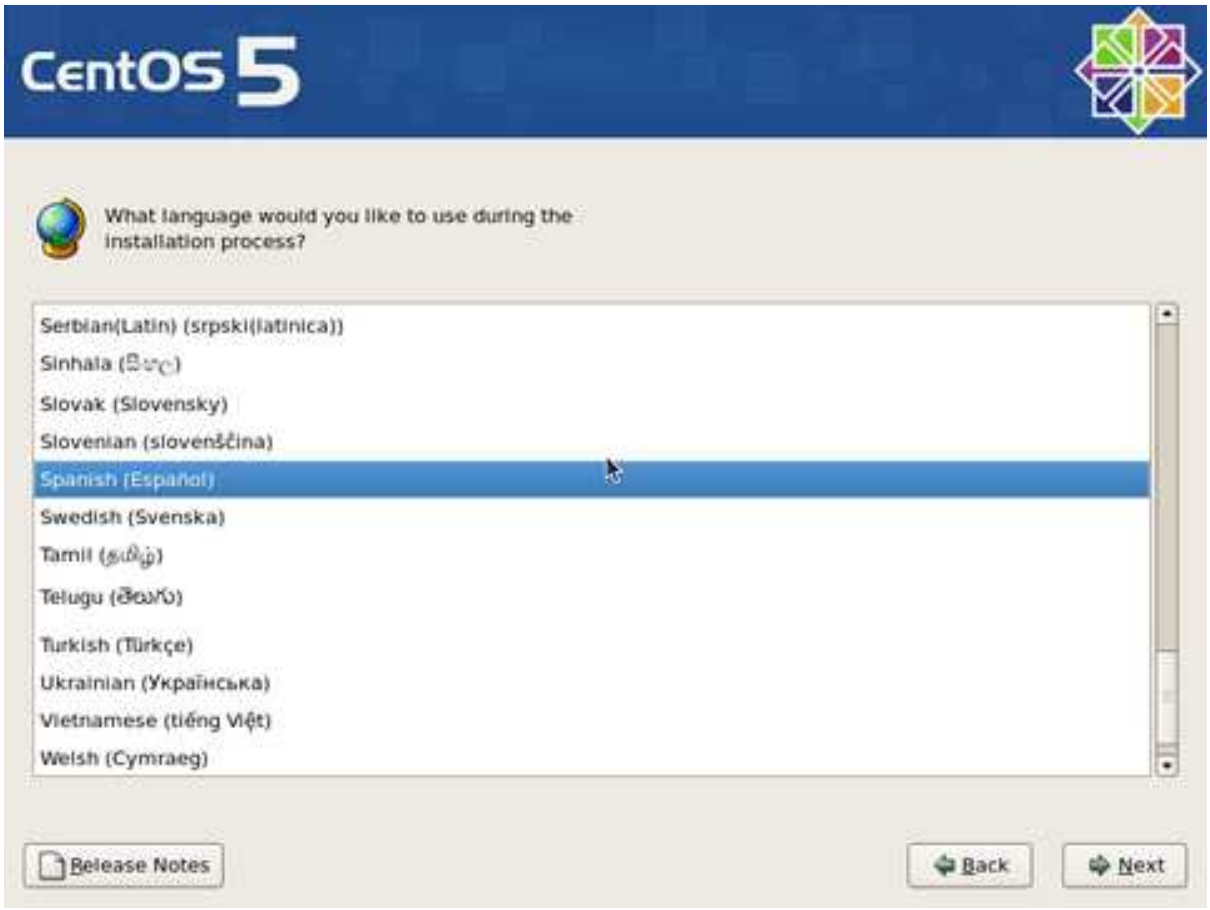


Figura 1.3: Ventana de Selección de Idioma

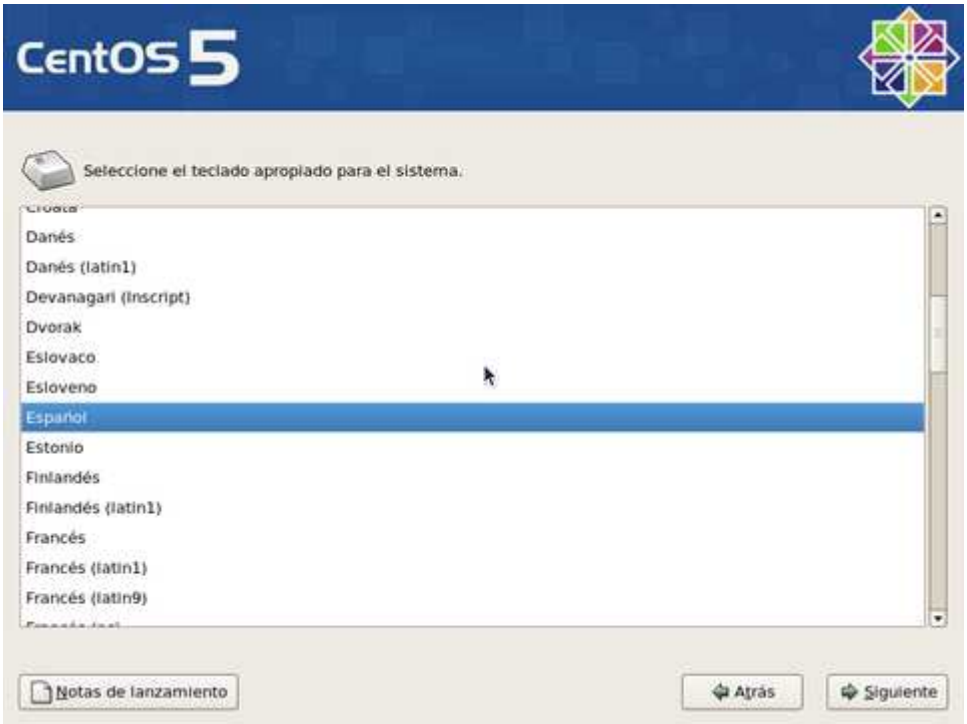


Figura 1.4: Ventana de Seleccion de Teclado

En la siguiente ventana (Figura 1.5), salvo que exista una instalación previa que se desee actualizar (no recomendado), deje seleccionado **«Instalar CentOS»** y haga clic en el botón **«Siguiente»** a fin de realizar una instalación nueva.

Figura 1.5: Opciones de Instalación

En la ventana de Opciones de Partición (Figura 1.6) se puede seleccionar:

«Remover particiones en dispositivos seleccionados y crear disposición», lo cual eliminaría cualquier partición de cualquier otro sistema operativo presente, y creará de forma automática las particiones necesarias.

«Remover particiones de linux en dispositivos seleccionados y crear disposición», lo cual eliminaría cualquier partición otra instalación de Linux presente, y creará de forma automática las particiones necesarias.

«Usar espacio disponible en dispositivos seleccionados y crear disposición», lo cual creará de forma automática las particiones necesarias en el espacio disponible.

Luego haga click en «Revise y modifique la capa de particiones» para poder ver el resultado final de particionado luego haga click en **«Siguiente»**

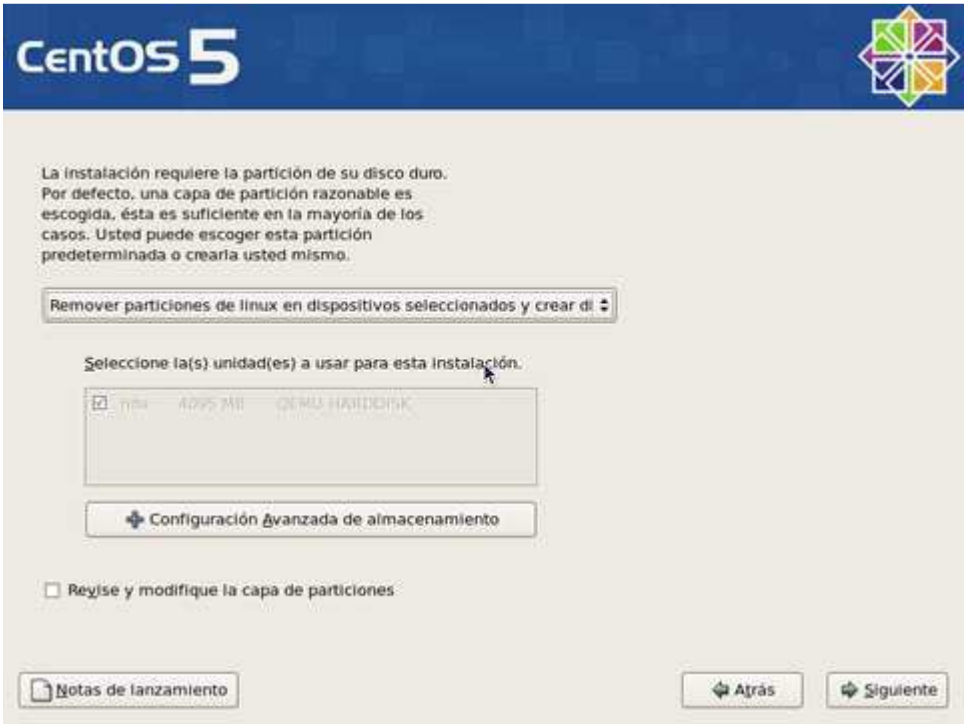


Figura 1.6: Opciones de Particionado de Disco Duro

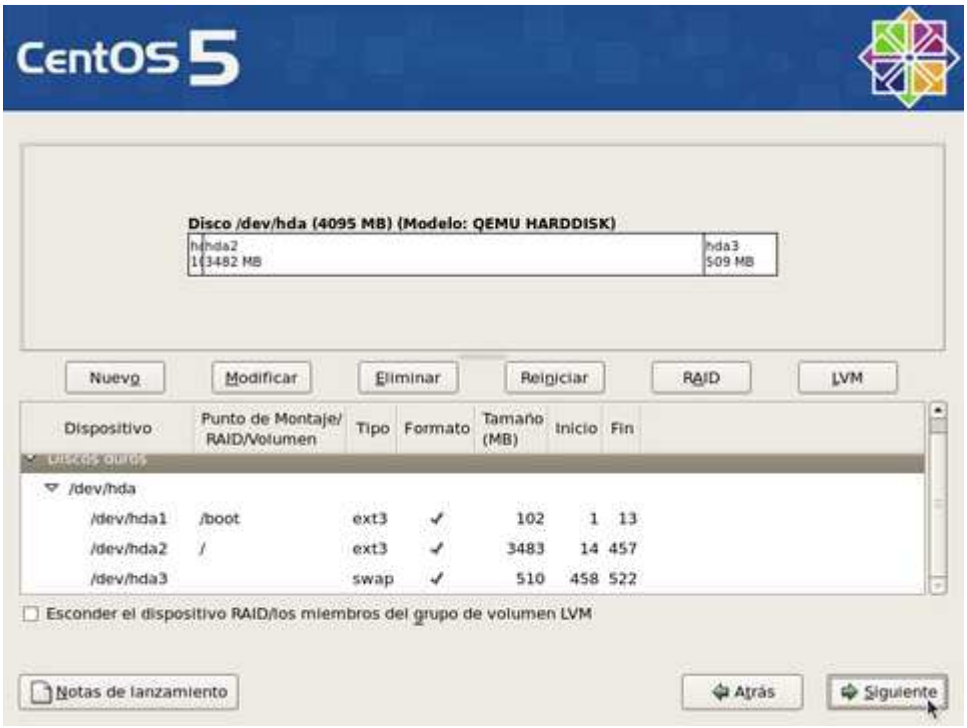


Figura 1.7: Tabla de Particiones

Si está conforme con la tabla de particiones creada (Figura 1.7), haga clic sobre el botón **«siguiente»**

Luego Ingresará a la configuración del gestor de arranque(Figura 1.8),por motivos de seguridad y principalmente con la finalidad de impedir que alguien sin autorización y con acceso físico al sistema pueda iniciar el sistema, haga clic en la casilla **«Usar la contraseña del gestor de arranque»**.

También puede configurar el sistema que arrancará por defecto (si tiene instalado algún otro), las etiquetas que se mostraran en el menú de arranque, etc… Cuando estemos de acuerdo con toda la configuración damos click en **«Siguiente»**.

En la siguiente ventana (Figura 1.9) se van a configurar los parámetros de red del sistema, haga clic sobre el botón **«Modificar»** o si quiere hacerlo después presione **«Siguiente»**.

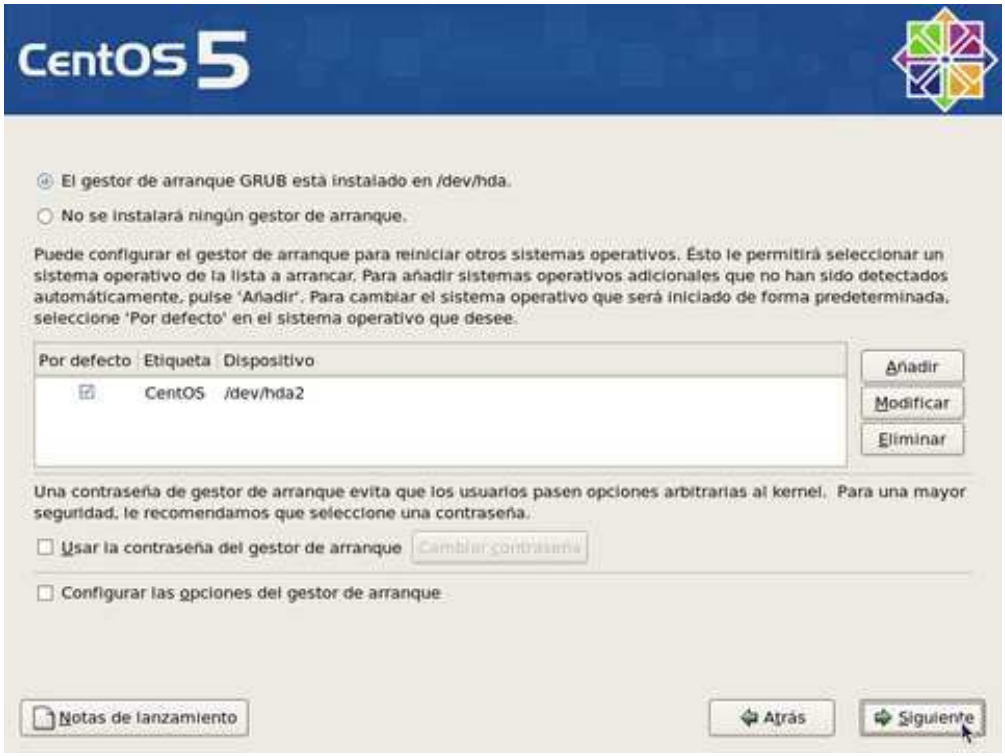


Figura 1.8: Configuración de Arranque

Figura 1.9: Configuración de Red

En la ventana de Zona Horaria(Figura 1.10) Seleccione la casilla **«El sistemahorario usará UTC»**, que significa que el reloj del sistema utilizará [1]**UTC** que es el sucesor de [2]**GMT** y es la zona horaria de referencia respecto a la cual se calculan todas las otras zonas del mundo. Haga clic con el ratón sobre la región que corresponda en el mapa mundial o seleccione en el siguiente campo la zona horaria que corresponda a la región donde se hospedará físicamente el sistema.

[1]**Tiempo Universal Coordinado**

[2] **Greenwich Mean Time(TiempoPromedio de Greenwich)**

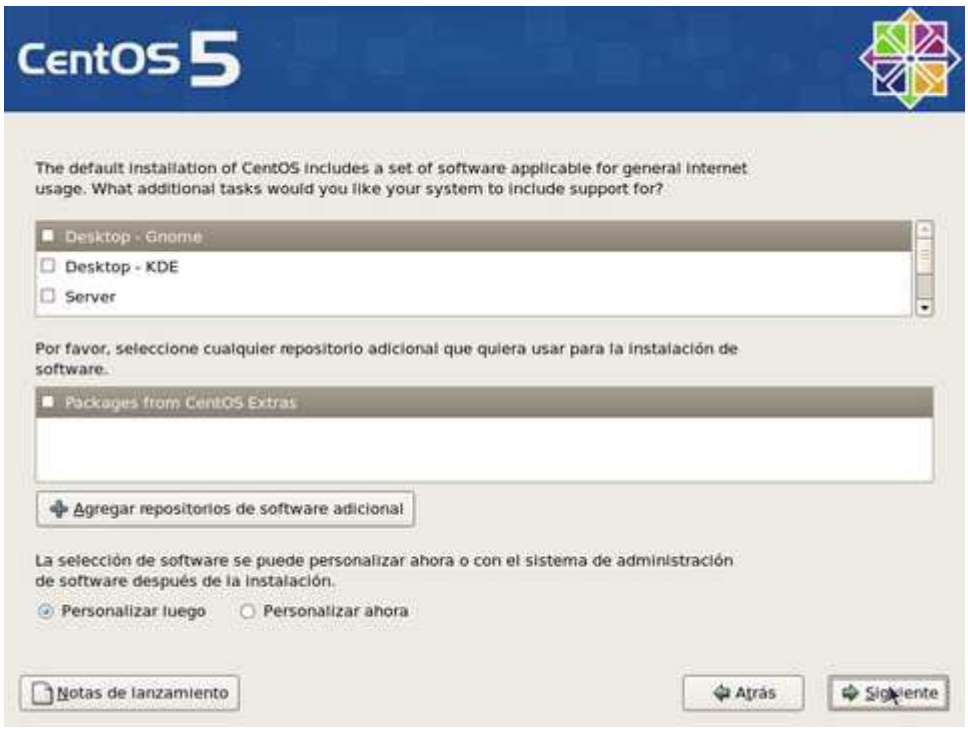
Figura 1.10: Selección de Zona Horaria

En la siguiente ventana (Figura 1.11) se va a asignar una clave de acceso al usuario **root**. Por razones de seguridad, se recomienda asignar una clave de acceso que evite utilizar palabras provenientes de cualquier diccionario, en cualquier idioma, así como cualquier combinación que tenga relación con datos personales. Una vez fijada la clave damos click en **«Siguiente»**.

Figura 1.11: Definición de Clave root

En la siguiente pantalla (Figura 1.12) podrá seleccionar los grupos de paquetes que quiera instalar en el sistema. Añada o elimine a su conveniencia. Lo recomendado, sobre todo si se trata de un servidor, es realizar una instalación con el mínimo de paquetes, desactivando todas las casillas para todos los grupos de paquetes. El objeto de esto es solo instalar lo mínimo necesario para el funcionamiento del sistema operativo, y permitir instalar posteriormente solo aquello que realmente se requiera de acuerdo a la finalidad productiva que tendrá el sistema. Al terminar, haga clic sobre el botón **«Siguiente»**.

Luego se realizará una comprobación de dependencias de los paquetes a instalar(Figura 1.13). Este proceso puede demorar algunos minutos.



t

Figura 1.12: Selección de Paquetes

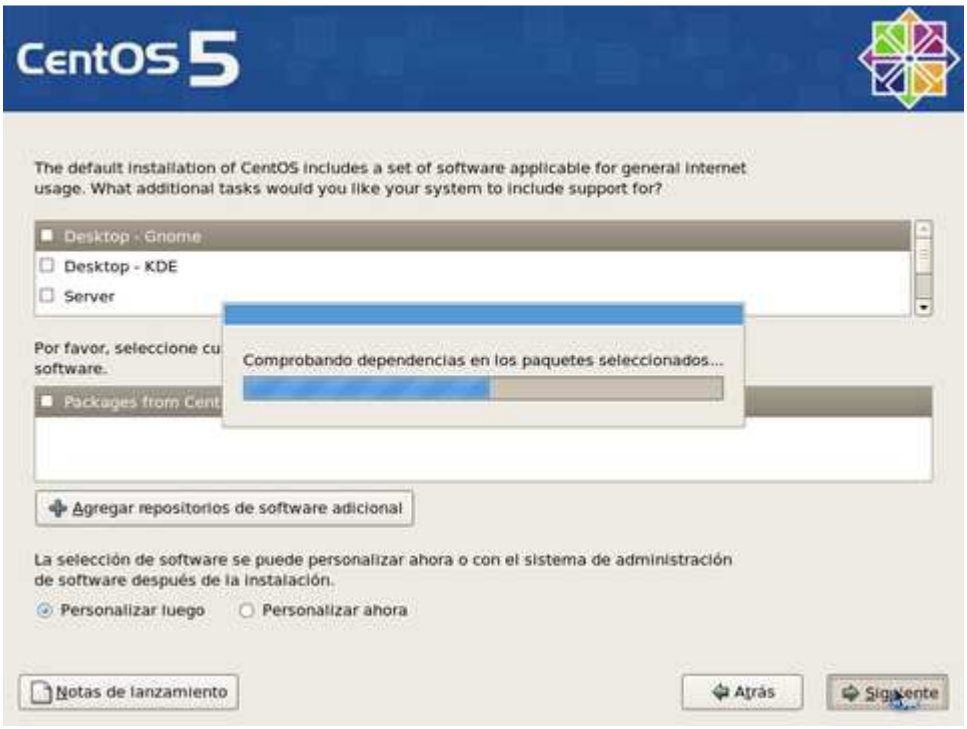


Figura 1.13: Comprobación de Dependencias

Antes de iniciar la instalación sobre el disco duro, el sistema le informará(Figura 1.14) respecto a que se guardará un registro del proceso en si en el fichero **/root/install.log**. Para continuar, haga clic sobre el botón **«Siguiente»**.

Figura 1.14: Información previa a la Instalación

Se iniciará de forma automática el proceso de formato de las particiones que haya creado para instalar el sistema operativo (Figura 1.15). Dependiendo de la capacidad del disco duro, este proceso puede demorar algunos minutos.

Se realizará automáticamente una copia de la imagen del programa de instalación (Figura 1.16) sobre el disco duro a fin de hacer más eficiente el proceso. Dependiendo de la capacidad del microprocesador y cantidad de memoria disponible en el sistema, este proceso puede demorar algunos minutos.

Espere a que se terminen los preparativos de inicio del proceso de instalación (Figura 1.17).

t 

Figura 1.15 Formateando Disco Duro

Figura 1.16: Copia de Imagen de Instalación

Figura 1.17: Preparativos de Instalación

Iniciará la instalación de los paquetes(Figura 1.18) necesarios para el funcionamiento del sistema operativo. Espere algunos minutos hasta que concluya el proceso.

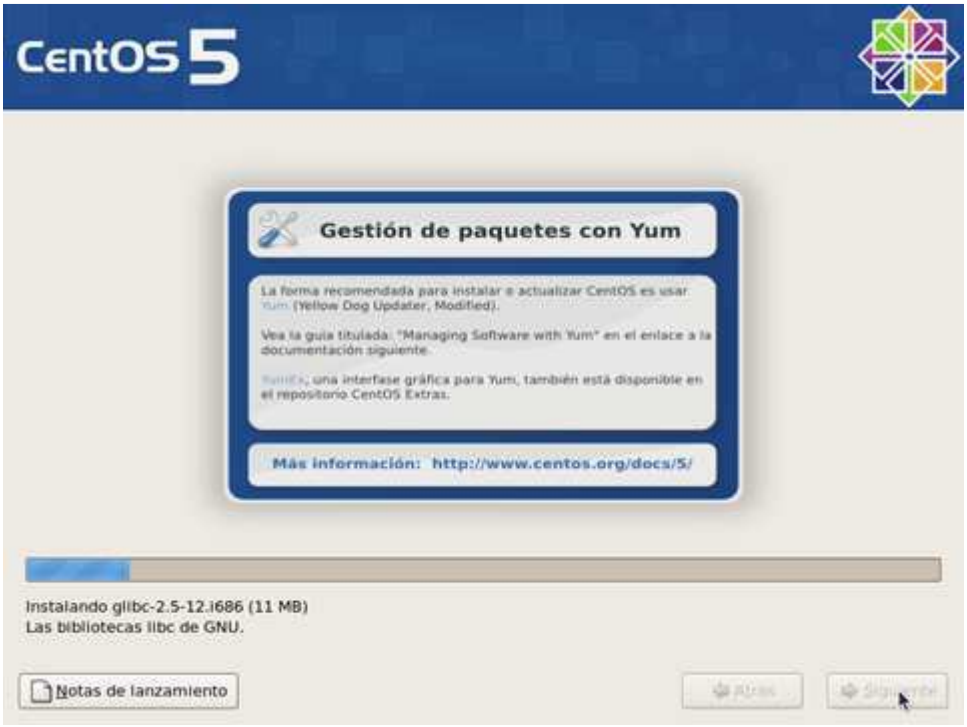


Figura 1.18: Instalación de Paquetes

Una vez concluida la instalación de los paquetes, haga clic sobre el botón **«Reiniciar»**(Figura 1.19).

Figura 1.19: Reiniciar Equipo

ANEXO 2 | CONFIGURACIÓN DE RED

2.1 CONFIGURACIÓN SERVIDOR DHCP

Ahora que tenemos nuestro servidor CentOS y Asterisk correctamente instalados procedemos a configurarlos a fin de poder Crear nuestra Central Telefónica.

Lo primero que vamos a configurar es el servidor [1]DHCP, ya que debemos brindar a los usuarios la facilidad de poder registrarse en cualquier equipo de la red y evitarnos configurar una dirección estática para cada equipo.

Para esto hacemos uso de cualquier editor de texto como nano y editamos el archivo de configuración:

```
[root@servidor ~]# nano /etc/dhcpd.conf
```

en nuestro caso se ha configurado de la siguiente forma:

```
subnet 192.168.10.0 netmask 255.255.255.0 {

        option subnet-mask    255.255.255.0;

        range 192.168.10.11 192.168.10.250;

        default-lease-time 21600;

        max-lease-time 43200;

}
```

Ahora definimos una dirección IP para el servidor desde la línea de comandos:

```
[root@servidor ~]# ifconfig eth0 192.168.10.10 netmask 255.255.255.0
```

Solo nos queda reiniciar e iniciar los servicios de red y DHCP respectivamente:

```
[root@servidor ~]# /etc/init.d/network restart

Interrupción de la interfaz eth0:                          [  OK  ]

Interrupción de la interfaz de loopback:                   [  OK  ]
```

[1]**Dynamic Host Control Protocol (Protocolo de Control Dinámico de Host)**

```
Activación de la interfaz de loopback:                          [  OK  ]

Activandointerfaz eth0:                                         [  OK  ]

[root@servidor ~]# ifconfig eth0

eth0      Link encap:EthernetHWaddr 00:1B:24:1E:C3:11

inet addr:192.168.10.10  Bcast:192.168.10.255  Mask:255.255.255.0

          UP BROADCAST MULTICAST  MTU:1500  Metric:1

          RX packets:0 errors:0 dropped:0 overruns:0 frame:0

          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0

collisions:0 txqueuelen:1000

          RX bytes:0 (0.0 b)  TX bytes:0 (0.0 b)

          Interrupt:169

[root@servidor ~]# /etc/init.d/dhcpd start

Iniciando dhcpd:                                                [  OK  ]
```

Con esto tenemos configurado nuestro servidor DHCP, solo resta comprobar en los equipos cliente si nos está entregando una dirección adecuada.

2.2 CONFIGURACIÓN DE RED IPV6

2.2.1 En clientes XP

Abrir la línea de comandos (Figura 2.1) tecleando cmd en la ventana Ejecutar(Inicio -> Ejecutar):

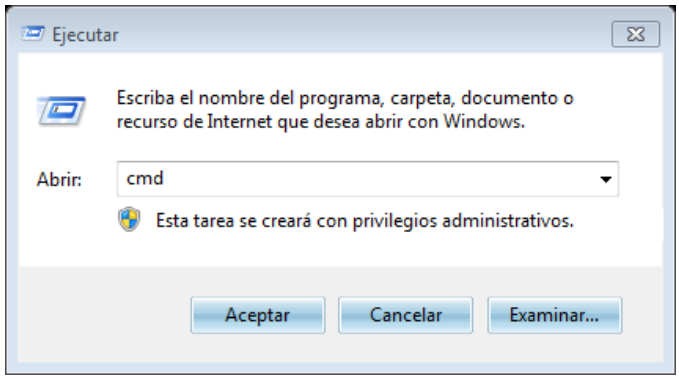


Figura 2.1: Ventana Ejecutar

Luego teclear las siguientes líneas

```
ipv6 install

netsh

interface ipv6 ad address “wired” fec0:0:0:f101::3

interface ipv6 set interfac “wired” site prefixlength=120

interface ipv6 add route ::/0 feco:0:0:f101::4
```

Donde “wired” es el nombre de la interfaz de redpara este caso, pero podemos renombrarlo con cualquier nombre. En la Figura 2.2 se muestra la interfaz antes y después de renombrarla.

Figura 2.2:

2.2.2 En clientes kubuntu

Abrir un terminal y ejecutar los siguientes commandos:

```
ipaddr add fec0:0:0:f101::3

dev eth0 route -Ainet6 add fec0:0:0:f101::4/64

dev eth0 route -Ainet6 add default  gw fec0:0:0:f101::4
```

2.2.3 En el Servidor CentOS

Abrir el archivo /etc/sysconfig/network con un editor de textos y editar las siguientes líneas

```
NETWORKING=yes
NETWORKING_IPV6=yes
HOSTNAME=localhost.localdomain
```

En el archivo de configuración de la interfaz /etc/sysconfig/network-scripts/ifcfg-eth0 agregar las siguientes líneas:

```
# Realtek Semiconductor Co., Ltd. RTL-8139/8139C/8139C+
DEVICE=eth0
BOOTPROTO=none
HWADDR=00:13:8F:B2:FE:A0
ONBOOT=yes
TYPE=Ethernet
USERCTL=no
IPV6INIT=yes
IPV6ADDR=fec0:0:0:f101::4/64
PEERDNS=yes
NETMASK=255.255.255.0
IPADDR=192.168.1.23
GATEWAY=192.168.1.20
```

También en la interfaz de loopback en /etc/sysconfig/network-scripts/ifcfg-lo:

```
DEVICE=lo
IPADDR=127.0.0.1
NETMASK=255.0.0.0
NETWORK=127.0.0.0
```

```
# If you're having problems with gated making 127.0.0.0/8 a martian,
# you can change this to something else (255.255.255.255, for example)
BROADCAST=127.255.255.255
ONBOOT=yes
NAME=loopback
IPV6INIT=yes
IPV6ADDR=::1
```

ANEXO 3 | ASTERISKV6

3.1 INSTALACIÓN

Lo primero que debemos hacer es ir a la página de descarga de asterisk v6y descargar el paquete **asteriskv6-20080107.tar.bz2:**

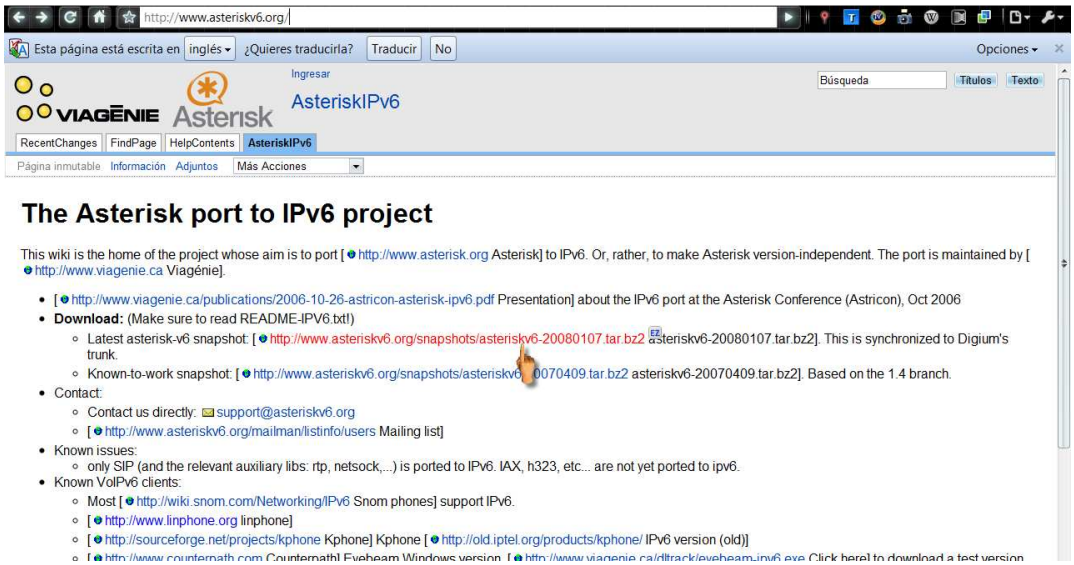


Figura 3.1: Página de descarga de Asteriskv6

Lo guardamos en cualquier carpeta

Antes de proceder a Instalar Asteriskv6 debemos asegurarnos de que en nuestro sistema cuenta con las siguientes de dependencias: bison,ncurses, zlib, openssl, gnutls-devel, gcc, gcc-c++.

En caso de que nos falte alguna podemos instalarlas fácilmente utilizando yum:

```
yuminstall [nombre dependencia]
```

Una vez comprobadas las dependencias ya podemos instalar asteriskv6 desde la carpeta donde lo descargamos ejecutando los siguientes comandos:

```
./configure

make
```

```
make install
```

3.2 CONFIGURACIÓN

Tenemos todo listo para empezar a configurar nuestra Central de Telefonía IP para esto hacemos uso de los archivos de configuración de Asterisk que se encuentran en /etc/asterisk/:

```
[root@servidor ~]# ls /etc/asterisk/
adsi.confextensions.conf.bkphoneprov.conf
adtranvofr.confextensions.luaqueuerules.conf
agents.confextensions_minivm.confqueues.conf
alarmreceiver.conffeatures.confres_ldap.conf
alsa.conffestival.confres_odbc.conf
amd.conffollowme.confres_pgsql.conf
asterisk.adsifunc_odbc.confres_snmp.conf
asterisk.confgtalk.confrpt.conf
cdr_adaptive_odbc.conf   h323.conf               rtp.conf
cdr.confhttp.confsay.conf
cdr_custom.confiax.confsip.conf
cdr_manager.confiaxprov.confsip.conf~
cdr_odbc.confindications.confsip.conf.bk
cdr_pgsql.confjabber.confsip_notify.conf
cdr_sqlite3_custom.conf  jingle.confskinny.conf
cdr_tds.conflogger.confsla.conf
chan_dahdi.confmanager.confsmdi.conf
cli.confmeetme.conf             telcordia-1.adsi
codecs.confmgcp.confudptl.conf
console.confminivm.confunistim.conf
dnsmgr.confmisdn.confusbradio.conf
dundi.confmodules.confusers.conf
enum.confmusiconhold.confvoicemail.conf
```

```
extconfig.confmuted.confvoicemail.conf~
extensions.aelosp.confvpb.conf
extensions.confoss.conf
extensions.conf~          phone.conf
```

3.2.1 Creación de Cuentas SIP (sip.conf)

Para registrar las cuentas de usuario debemos modificar el archivo sip.conf, en este archivo configuraremos las características más básicas de nuestras extensiones SIP, para esto utilizamos cualquier editor de texto

```
[2001]
type=friend
username=2001
callerid="Usuario 1"
secret=c2001
host=dynamic
context=central4

[2002]
type=friend
username=2002
callerid="Usuario 2"
secret=c2002
host=dynamic
context=central4

[2003]
type=friend
username=2003
callerid="Usuario 3"
secret=c2003
```

```
host=dynamic

context=central4


[2004]

type=friend

username=2004

callerid="Usuario 4"

secret=c2004

host=dynamic

context=central4
```

La información importante es el nombre o número de la extensión, encerrada entre corchetes, el tipo de cuenta (type) que puede ser friend|user|peer, la contraseña del usuario(secret), el contexto (context) o grupo de extensiones al que pertenecerá esta cuenta y la etiqueta para identificación del llamante (callerid). Estos son los datos básicos de usuario, existen otras opciones que se pueden ir agregando según los servicios que se requiera en nuestra central telefónica.

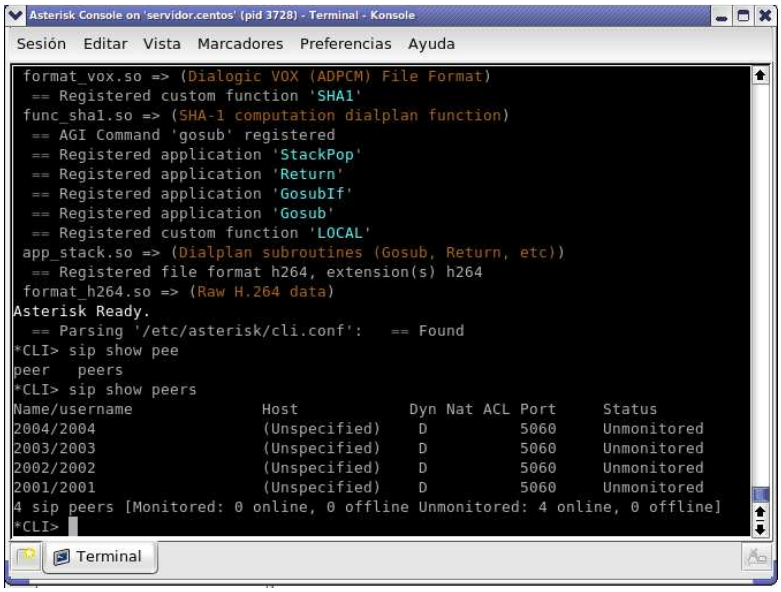


Figura 3.2: Comprobación de cuentas SIP

Ahora reiniciamos la configuración sip y comprobamos que las cuentas estén registradas (Figura 3.2).

3.2.2 Dialplan (extensions.conf)

Este archivo permite tener la configuración de las extensiones y para empezar detallaremos en que consiste y como está conformado este archivo.

Por lo pronto vamos a crear un plan de marcado para que los usuarios puedan hacer llamadas entre sí, luego seguiremos agregando más funcionalidades, entonces procedemos a abrir el archivo antes mencionado y agregamos las siguientes líneas al final del archivo:

```
[central4]
exten => 2001,1,Dial(SIP/2001,30,m)
exten => 2001,2,Hangup()

exten => 2002,1,Dial(SIP/2002,30,m)
exten => 2002,2,Hangup()

exten => 2003,1,Dial(SIP/2003,30,m)
exten => 2003,2,Hangup()

exten => 2004,1,Dial(SIP/2004,30,m)
exten => 2004,2,Hangup()
```

En este caso al llamar a la extensión 2001(o cualquier otra) usamos el comando *Dial (destino, tiempo de timeout, opciones)*

El destino es el usuario 2001 del archivo sip.conf, 30 segundos de timeout, es decir que luego de 30 segundos si el usuario no contesta pasa a la siguiente prioridad. Las opciones hacen referencia a opciones del comando dial: la "m" indica que vamos a oír una música especial mientras esperamos a que el otro usuario conteste.

3.2.3 Voces en Español

En este momento tenemos configurados nuestros usuarios y un dialplan básico que nos permitirá hacer llamadas entre los usuarios registrados, pero por el momento el único idioma instalado es el inglés, lo podríamos dejar así pero para facilidad de los usuarios de nuestra central vamos a proceder a instalar las voces en español.

Todos los sonidos de asterisk se guarda por defecto en la carpeta /var/lib/asterisk/sounds, por tanto vamos a revisar el contenido de esta carpeta:

```
[root@servidor ~]# ls /var/lib/asterisk/sounds

en
```

Como podemos observar tenemos una sola carpeta "en" donde están los sonidos en inglés. Para instalar los sonidos en español lo primero que hacemos es descargarlos de los siguientes links:

http://www.asterio.com.ar/resources/downloads/ThaisaC-core-sounds-sln-1.4.12.tar.gz

http://www.asterio.com.ar/resources/downloads/ThaisaC-extra-sounds-sln-1.4.12.tar.gz

y los guardamos en cualquier carpeta, en este caso vamos a guardar en /tmp/es:

```
[root@servidor es]# cd /tmp/es
```

```
ThaisaC-core-sounds-sln-1.4.12.tar.gzThaisaC-extra-sounds-sln-1.4.12.tar.gz
```

El siguiente paso es descomprimir los dos archivos:

```
[root@servidores]# tar -xfvThaisaC-core-sounds-sln-1.4.12.tar.gzThaisaC-extra-sounds-sln-1.4.12.tar.gz
```

Una vez descomprimidos los archivos podemos eliminar los comprimidos a fin de que queden solo los archivos de sonido dentro de la carpeta "es" y entonces moverla a la carpeta de asterisk con el siguiente comando:

```
[root@servidor ~]# mv /tmp/es /var/lib/asterisk/sounds/es
```

Solo nos queda revisar que se haya movido la carpeta correctamente:

```
[root@servidor ~]# ls /var/lib/asterisk/sounds

enes
```

Por ultimo tenemos que hacer que definir el idioma que van a utilizar nuestros usuarios editando el archivo sip.conf agregando la siguiente línea:

```
[2001]

type=friend

username=2001

callerid="Usuario 1"

secret=c2001

host=dynamic

language=es

context=central4


[2002]

type=friend

username=2002

callerid="Usuario 2"

secret=c2002

host=dynamic
```

```
language=es
context=central4


[2003]
type=friend
username=2003
callerid="Usuario 3"
secret=c2003
host=dynamic
language=es
context=central4


[2004]
type=friend
username=2004
callerid="Usuario 4"
secret=c2004
host=dynamic
language=es


context=central4
```

3.2.4 Correo de Voz (voicemail.conf)

Uno de los servicios más importantes que podemos ofrecer en nuestra central telefónica es el buzón de voz, el archivo voicemail.confsirve para configurar el contestador automático y gestionar los buzones de los usuarios.

Hemos modificado las opciones emailsubject y emailbody para que cuando se envíe el mensaje el asunto y el cuerpo del mismo estén personalizados a nuestro gusto y además agregamos buzones de correo para cada usuario.

```
[general]
format=gsm|wav
attach=yes
maxmsg=100
maxsecs=180
minsecs=3
skipms=3000
maxsilence=10
silencethreshold=128
maxlogins=3
moveheard=yes
pbxskip=yes
fromstring=Asterisk PBX
emailsubject=[PBX]: Nuevo mensaje ${VM_MSGNUM} en el buzon: ${VM_MAILBOX}
emailbody=Estimado ${VM_NAME}:\n\n\tLeinformamosque ha recibido un mensaje en subuzon.\n\n\tDatos:\n\n\tDuraciÃ³n: ${VM_DUR}\n\tNumero: ${VM_MSGNUM}\n\tBuzon: ${VM_MAILBOX}\n\tRemitente: ${VM_CALLERID}\n\tFecha: ${VM_DATE}\n\n\t Atentamente.\n\n\t\t\t\t--Asterisk\n
emaildateformat=%A, %B %d, %Y at %r
mailcmd=/usr/sbin/sendmail -t
<mailbox>=<password>,<name>,<email>,<pager_email>,<options>
attach=yes
attachfmt=wav
saycid=yes
sendvoicemail=yes
review=yes
forcename=yes
forcegreetings=no
hidefromdir=yes
tempgreetwarn=yes


listen-control-forward-key=#
```

```
listen-control-reverse-key=*

listen-control-pause-key=0

listen-control-restart-key=2

listen-control-stop-key=13456789

backupdeleted=100


[zonemessages]

eastern=America/New_York|'vm-received' Q 'digits/at' IMp

central=America/Chicago|'vm-received' Q 'digits/at' IMp

central24=America/Chicago|'vm-received' q 'digits/at' H N 'hours'

military=Zulu|'vm-received' q 'digits/at' H N 'hours' 'phonetic/z_p'

european=Europe/Copenhagen|'vm-received' a d b 'digits/at' HM


[default]

;1234 => 4242,Example Mailbox,root@localhost

;4200 => 9855,Mark Spencer,markster@linux-
support.net,mypager@digium.com,attach=no|serveremail=myaddy@digium.com|tz=cent
ral|maxmsg=10

;4300 => 3456,Ben Rigas,ben@american-computer.net

;4310 => -5432,Sales,sales@marko.net

;4069 => 6522,Matt
Brooks,matt@marko.net,,|tz=central|attach=yes|saycid=yes|dialout=fromvm|callba
ck=fromvm|review=yes|operator=yes|envelope=yes|moveheard=yes|sayduration=yes|s
aydurationm=1

;4073 => 1099,Bianca Paige,bianca@biancapaige.com,,delete=1

;4110 => 3443,Rob Flynn,rflynn@blueridge.net

;4235 => 1234,Jim Holmes,jim@astricon.ips,,Tz=european


301=>111,Usuario1,usuario1@dominio1.com

302=>222,Usuario2,usuario2@dominio2.com

303=>333,Usuario3,usuario3@dominio3.com

304=>444,Usuario4,usuario4@dominio4.com
```

También necesitamos editar los archivos sip.conf y extensions.conf:

3.2.4.1 sip.conf

En el archivosip.conf, basta con especificar el número del buzón de voz y el contexto donde esté definido en voicemail.conf, en nuestro caso están en el contexto default.

```
[2001]
type=friend
username=2001
callerid="Usuario 1"
secret=c2001
mailbox=301@default
host=dynamic
language=es
context=central4

[2002]
type=friend
username=2002
callerid="Usuario 2"
secret=c2002
mailbox=302@default
host=dynamic
language=es
context=central4

[2003]
type=friend
username=2003
```

```
callerid="Usuario 3"
secret=c2003
mailbox=303@default
host=dynamic
language=es
context=central4

[2004]
type=friend
username=2004
callerid="Usuario 4"
secret=c2004
mailbox=304@default
host=dynamic
language=es
context=central4
```

3.2.4.2 extensions.conf

En el archivo extensions.confse necesitan algunas líneas extras para que funcione el buzón de voz, lo primero que podemos observar es que en lugar de utilizar números en la definición de prioridades se utiliza la letra "n" que simplemente representa la siguiente prioridad(next), también se han agregado las prioridades de ocupado (101+1) y de no disponible (201+1) que lo que harán es llamar al buzón de voz si el usuario no contesta o no está conectado, la opción "Playback" lo único que va a hacer es reproducir un mensaje de despedida luego de haber dejado el mensaje, por último se agregó la extensión 3000 que será el número por el cual los usuarios pueden revisar sus mensajes.

```
[central4]


exten => 2001,1,Dial(SIP/2001,30,m)

exten => 2001,n,VoiceMail(301@default)

exten => 2001,n,Playback(vm-goodbye)

exten => 2001,n,Hangup()

exten =>2001,102,VoiceMail(301@default)

exten => 2001,103,Playback(vm-goodbye)

exten => 2001,104,Hangup()

exten => 2001,202,VoiceMail(301@default)

exten => 2001,203,Playback(vm-goodbye)

exten => 2001,204,Hangup()




exten => 2002,1,Dial(SIP/2002,30,m)

exten => 2002,n,VoiceMail(302@default)

exten => 2002,n,Playback(vm-goodbye)

exten => 2002,n,Hangup()

exten =>2002,102,VoiceMail(302@default)

exten => 2002,103,Playback(vm-goodbye)

exten => 2002,104,Hangup()

exten =>2002,202,VoiceMail(302@default)

exten => 2002,203,Playback(vm-goodbye)

exten => 2002,204,Hangup()




exten => 2003,1,Dial(SIP/2003,30,m)

exten => 2003,n,VoiceMail(303@default)

exten => 2003,n,Playback(vm-goodbye)

exten => 2003,n,Hangup()

exten => 2003,102,VoiceMail(303@default)
```

```
exten => 2003,103,Playback(vm-goodbye)

exten => 2003,104,Hangup()

exten =>2003,202,VoiceMail(303@default)

exten => 2003,203,Playback(vm-goodbye)

exten => 2003,204,Hangup()


exten => 2004,1,Dial(SIP/2004,30,m)

exten => 2004,n,VoiceMail(304@default)

exten => 2004,n,Playback(vm-goodbye)

exten => 2003,n,Hangup()

exten =>2004,102,VoiceMail(304@default)

exten => 2004,103,Playback(vm-goodbye)

exten => 2003,104,Hangup()

exten =>2004,202,VoiceMail(304@default)

exten => 2004,203,Playback(vm-goodbye)

exten => 2004,204,Hangup()


exten => 3001,1,VoicemailMain(301@default)

exten => 3002,1,VoicemailMain(302@default)

exten => 3003,1,VoicemailMain(303@default)

exten => 3004,1,VoicemailMain(304@default)
```

ANEXO 4 | LINPHONE

4.1 INSTALACIÓN

4.1.1 Instalación en Distribuciones Debian como Ubuntu

La instalación en Kubuntu se puede hacer fácilmente a través del gestor de paquetes aptitude con el siguiente comando:

```
osvaldo@kubuntuOsvaldo:~$ sudo aptitudeinstall -y linphone

Leyendo lista de paquetes... Hecho

Creando árbol de dependencias

Leyendo la información de estado... Hecho

Leyendo la información de estado extendido

Inicializando el estado de los paquetes... Hecho

Escribiendo información de estado extendido... Hecho

Se instalarán los siguiente paquetes NUEVOS:

libavcodec52{a} libavutil49{a} libexosip2-4{a} libgsm1{a} liblinphone3{a}
libmediastreamer0{a} liboil0.3{a} libortp8{a} libosip2-4{a}

libreadline5{a} libschroedinger-1.0-0{a} libspeexdsp1{a} libswscale0{a}
linphonelinphone-common{a} linphone-nox{a}

0 paquetes actualizados, 16 nuevos instalados, 0 para eliminar y 0 sin
actualizar.

Necesito descargar 0B/11,5MB de archivos. Después de desempaquetar se usarán
23,1MB.

Escribiendo información de estado extendido... Hecho

Seleccionando el paquete libavutil49 previamente no seleccionado.

(Leyendo la base de datos ...  00%

91188 ficheros y directorios instalados actualmente.)

Desempaquetando libavutil49 (de .../libavutil49_4%3a0.5.1-1ubuntu1_i386.deb)
...

Seleccionando el paquete libgsm1 previamente no seleccionado.

Desempaquetando libgsm1 (de .../libgsm1_1.0.13-3_i386.deb) ...

Seleccionando el paquete liboil0.3 previamente no seleccionado.

Desempaquetando liboil0.3 (de .../liboil0.3_0.3.16-1ubuntu2_i386.deb) ...
```

```
Seleccionando el paquete libschroedinger-1.0-0 previamente no seleccionado.
Desempaquetando libschroedinger-1.0-0 (de .../libschroedinger-1.0-0_1.0.9.is.1.0.8-0ubuntu1_i386.deb) ...
Seleccionando el paquete libavcodec52 previamente no seleccionado.
Desempaquetando libavcodec52 (de .../libavcodec52_4%3a0.5.1-1ubuntu1_i386.deb) ...
Seleccionando el paquete libosip2-4 previamente no seleccionado.
Desempaquetando libosip2-4 (de .../libosip2-4_3.3.0-1_i386.deb) ...
Seleccionando el paquete libexosip2-4 previamente no seleccionado.
Desempaquetando libexosip2-4 (de .../libexosip2-4_3.3.0-1_i386.deb) ...
Seleccionando el paquete libortp8 previamente no seleccionado.
Desempaquetando libortp8 (de .../libortp8_3.2.1-1ubuntu4_i386.deb) ...
Seleccionando el paquete libspeexdsp1 previamente no seleccionado.
Desempaquetando libspeexdsp1 (de .../libspeexdsp1_1.2~rc1-1ubuntu1_i386.deb) ...
Seleccionando el paquete libswscale0 previamente no seleccionado.
Desempaquetando libswscale0 (de .../libswscale0_4%3a0.5.1-1ubuntu1_i386.deb) ...
Seleccionando el paquete libmediastreamer0 previamente no seleccionado.
Desempaquetando libmediastreamer0 (de .../libmediastreamer0_3.2.1-1ubuntu4_i386.deb) ...
Seleccionando el paquete liblinphone3 previamente no seleccionado.
Desempaquetando liblinphone3 (de .../liblinphone3_3.2.1-1ubuntu4_i386.deb) ...
Seleccionando el paquete libreadline5 previamente no seleccionado.
Desempaquetando libreadline5 (de .../libreadline5_5.2-7build1_i386.deb) ...
Seleccionando el paquete linphone-common previamente no seleccionado.
Desempaquetando linphone-common (de .../linphone-common_3.2.1-1ubuntu4_all.deb) ...
Seleccionando el paquete linphone-nox previamente no seleccionado.
Desempaquetando linphone-nox (de .../linphone-nox_3.2.1-1ubuntu4_i386.deb) ...
Seleccionando el paquete linphone previamente no seleccionado.
Desempaquetando linphone (de .../linphone_3.2.1-1ubuntu4_i386.deb) ...
```

```
Procesando disparadores para man-db ...
Configurando libavutil49 (4:0.5.1-1ubuntu1) ...

Configurando libgsm1 (1.0.13-3) ...

Configurando liboil0.3 (0.3.16-1ubuntu2) ...

Configurando libschroedinger-1.0-0 (1.0.9.is.1.0.8-0ubuntu1) ...

Configurando libavcodec52 (4:0.5.1-1ubuntu1) ...

Configurando libosip2-4 (3.3.0-1) ...

Configurando libexosip2-4 (3.3.0-1) ...

Configurando libortp8 (3.2.1-1ubuntu4) ...

Configurando libspeexdsp1 (1.2~rc1-1ubuntu1) ...

Configurando libswscale0 (4:0.5.1-1ubuntu1) ...

Configurando libmediastreamer0 (3.2.1-1ubuntu4) ...

Configurando liblinphone3 (3.2.1-1ubuntu4) ...

Configurando libreadline5 (5.2-7build1) ...

Configurandolinphone-common (3.2.1-1ubuntu4) ...
Configurando linphone-nox (3.2.1-1ubuntu4) ...
Configurando linphone (3.2.1-1ubuntu4) ...
```

```
Procesando disparadores para libc-bin ...
ldconfig deferred processing now taking place
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Leyendo la información de estado extendido
Inicializando el estado de los paquetes... Hecho
Escribiendo información de estado extendido... Hecho
osvaldo@kubuntuOsvaldo:~$
```

4.1.2 Instalación en Windows

- ✓ El primer paso para la instalación de este softphone es la descarga del instalador desde la página principal de linphone en el área de descargas para Windows:

http://download.savannah.gnu.org/releases-noredirect/linphone/stable/win32/

el paquete a descargar es linphone-3.3.1-setup.exe, ver Figura 4.1

Figura 4.1: Paquete de Instalación Linphone

- ✓ Hacer doble click en el paquete, luego nos pedirá elegir el idioma de instalación (Figura 4.2), elegimos español y damos click en Aceptar.
- ✓ Luego se ejecuta el Astente de instalación del paquete el cual orienta

al usuario en la instalación del softphone (Figura 4.3), damos cliclk en siguiente:

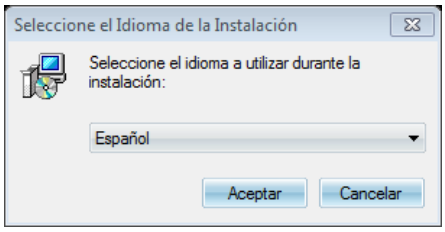


Figura 4.2: Selección de Idioma

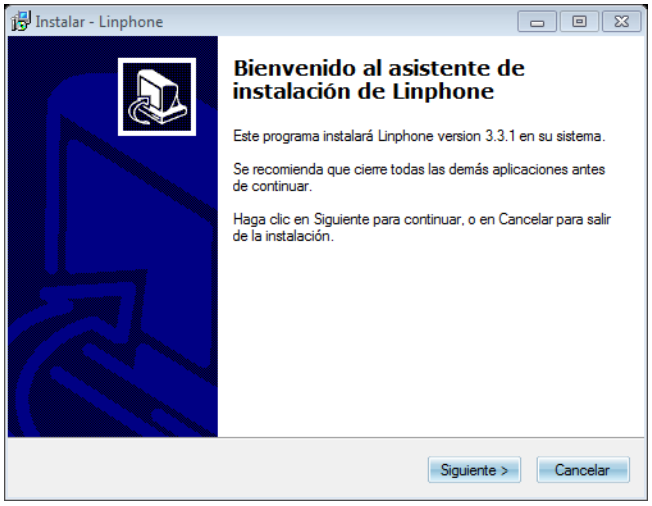


Figura 4.3: Ventaa inicial del Asistente

✓ El siguiente paso es aceptar la licencia del producto la cual es del tipo GNU GENERAL PUBLIC LICENSE (ver Figura 4.4), luego click en siguiente:

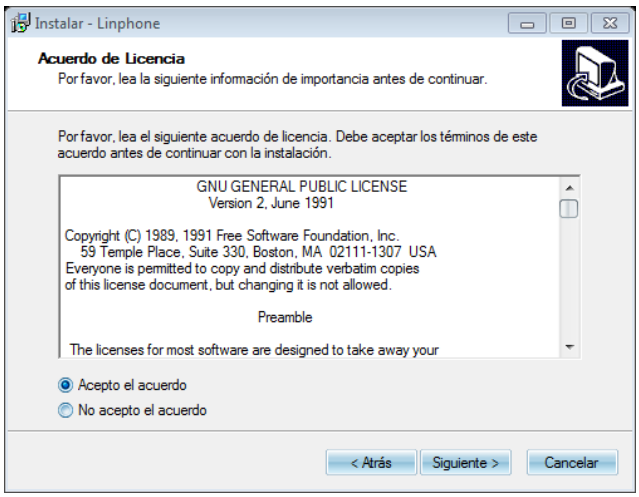


Figura 4.4: Acuerdo de Licencia

✓ Luego nos pide elegir la carpeta donde vamos instalar el softphone (ver Figura 4.5), damos click en siguiente:

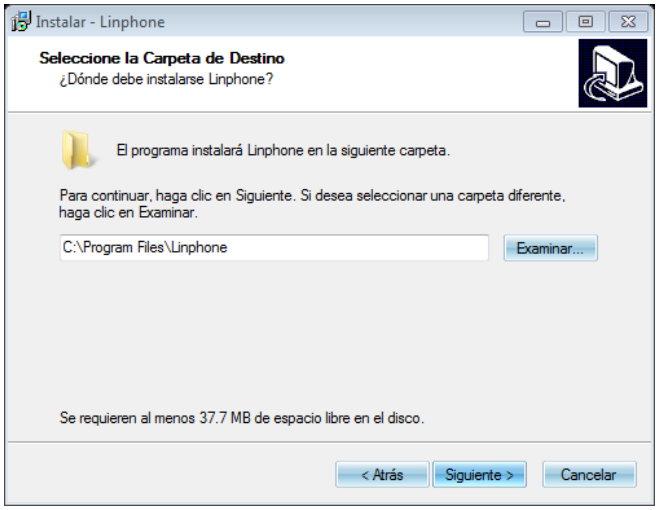


Figura 4.5: Elegir carpeta de Instalación de Linphone

✓ Luego aparecerá un resumen con las opciones que hemos elegido (Figura 4.6), le damos click en Instalar:

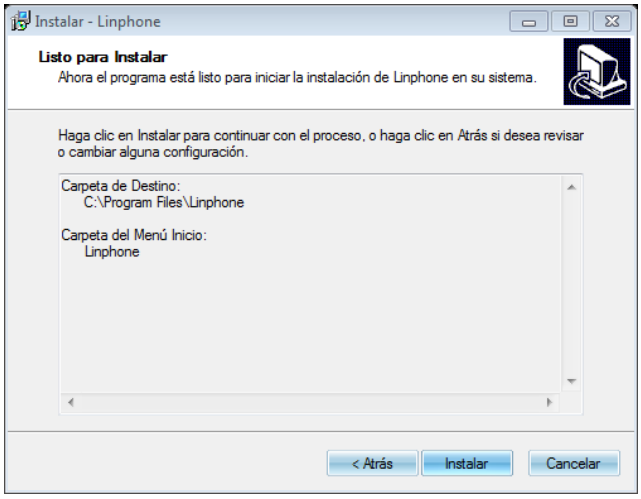


Figura 4.6: Resumen previo a la Instalación de Linhone

✓ Esperamos a que se instale… (ver Figura 4.7)

- ✓ Una vez que la instalación finaliza seleccionar Ejecutar Linphone (Figura 4.8) y hacer click en Finalizar:

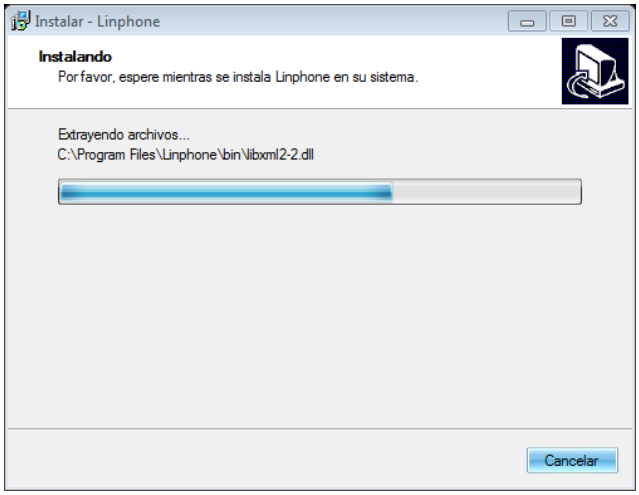


Figura 4.7: Instalando Linphone

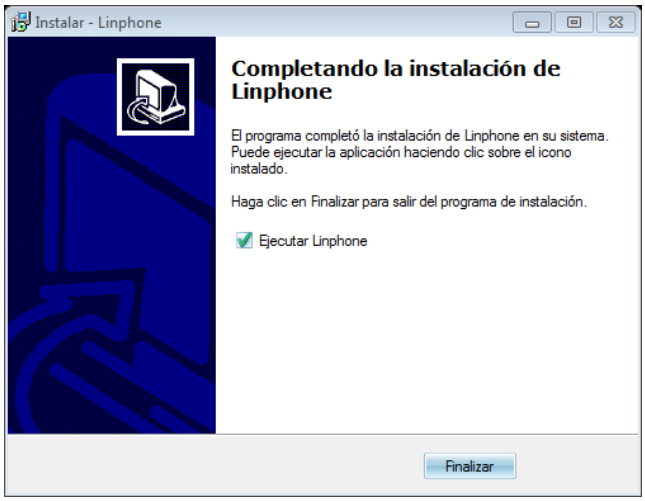


Figura 4.8: Finalizando la Instalación de Linphone

4.2 CONFIGURACIÓN

- ✓ Se ejecuta el softphone y aparece la ventana principal (ver Figura 4.9).
- ✓ Para configurar se debe ir al menú Linphone del softphone y escoger preferences (ver Figura 4.10).

- ✓ Luego Aparecerá la ventana que se muestra en la figura 4.11, aquí se habilita la opción que hace posible que este softphone trabajesobre IPv6**«Use IPv6 instead of IPv4»**, en caso de trabajar con IPv4 no se selecciona esta opción.

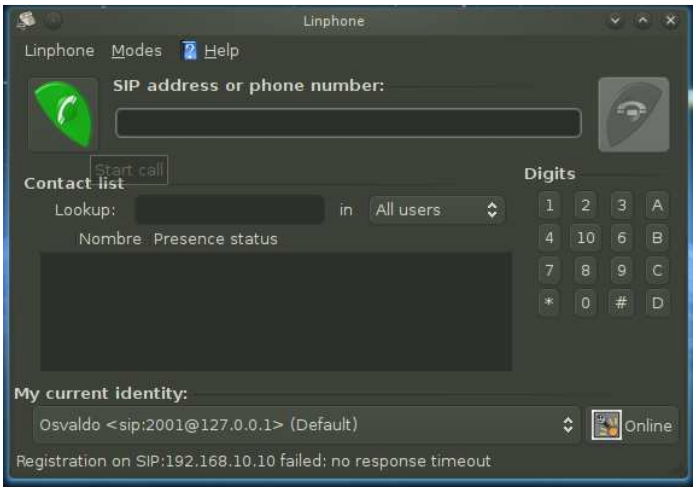


Figura 4.9: Ventana Principal de Linphone

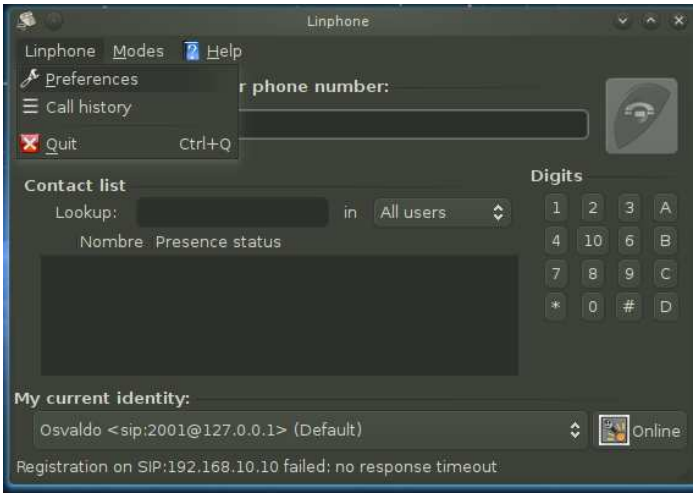


Figura 4.10: Menú Linphone

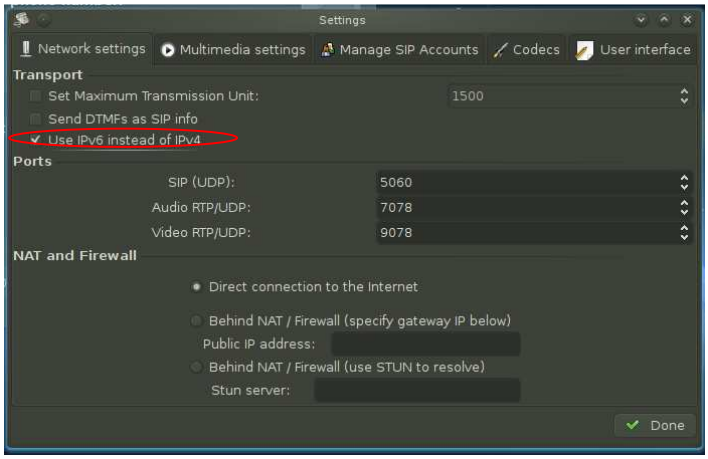


Figura 4.11: Network Settings

✓ El siguiente paso es dirigirse a la pestaña de Manage SIP Accounts (ver

Figura 4.12) en lasección de proxyaccounts se da click sobre el botón Add para configurar ladirección del servidor:

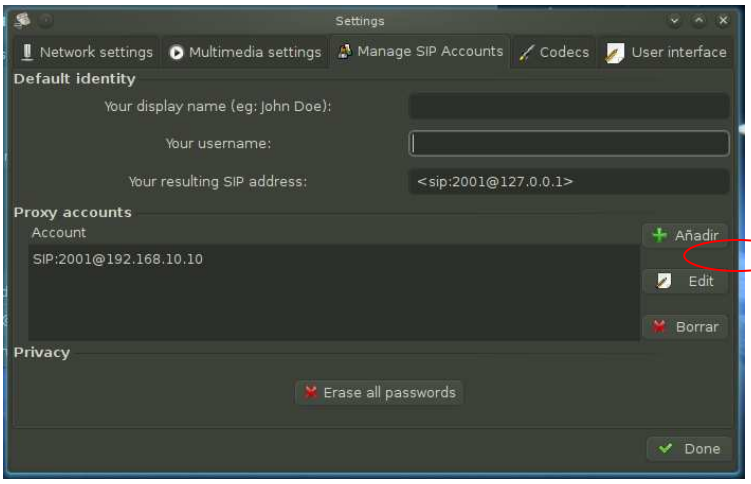


Figura 4.12: ManageSipAccounts

- ✓ En la ventana que aparece se configura la dirección IP del servidor en el cual se va a registrar elsoftphone ya sea con IPv4(Fi gura 4.13) o IPv6(Figura 4.14):

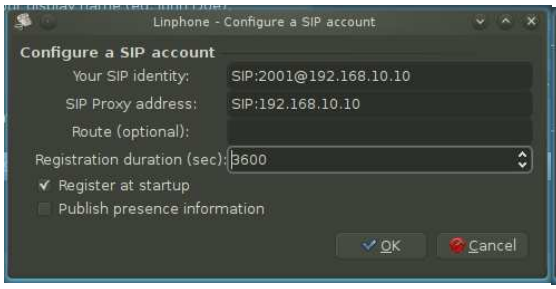


Figura 4.13: Cuenta SIP IPv4

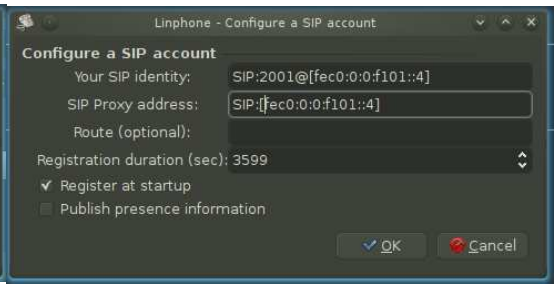


Figura 4.14:Cuenta SIP IPv6

- ✓ Luego nos pedira que nos autentiquemos (Figura 4.15)siempre y cuando tengamos ya configuradas las cuentas en un servidor.

Figura 4.15: Autenticación de Usuario en Linphone

ANEXO 5 | WIRESHARK

Wireshark es un capturador/analizador de paquetes de red (llamado a veces, sniffer o esnifer). Wireshark te permitirá ver, aun nivel bajo y detallado, qué está pasando en tu red. Además es gratuito, open source, y multiplataforma. Sin duda la mejor opción al momento de auditar nuestra red.

Posee una interfaz gráfica y muchas opciones de organización y filtrado de información. Así, permite ver todo el tráfico que pasa a través de una red (usualmente una red Ethernet, aunque es compatible con algunas otras)

5.1 ¿PARA QUÉ/QUIÉN ES ÚTIL WIRESHARK?

- ✓ Administradores lo usan para resolver problemas en la red
- ✓ Ingenieros lo usan para examinar problemas de seguridad
- ✓ Desarrolladores lo usan para depurar la implementación de los protocolos de red
- ✓ Estudiantes los usan para aprender internamente cómo funciona una red

5.2 CARACTERÍSTICAS DE WIRESHARK

- ✓ Disponible para Linux y Windows
- ✓ Captura de paquetes en vivo desde una intefaz de red
- ✓ Muestra los paquetes con información detallada de los mismos
- ✓ Abre y guarda paquetes capturados
- ✓ Importar y exportar paquetes en diferentes formatos
- ✓ Filtrado de información de paquetes
- ✓ Resaltado de paquetes dependiendo el filtro
- ✓ Crear estadísticas

5.3 INSTALACIÓN DE WIRESHARK

5.3.1 Instalación del Wireshark desde el tarball

Antes que nada, para poder compilar correctamente Wireshark debes tener dos cosas:

- ✓ Gtk+ y Glib, que puedes descargar de www.gtk.org
- ✓ libpcap, las librerías para captura de paquetes que Wireshark usa. La puedes encontrar en www.tcpdump.org

Ahora, debes descargar el código fuente de la página oficial(www.wireshark.opg/download.html) descomprimirlo e instalarlo:

```
tarzxvf wireshark-1.2.9.tar.gz
cd wireshark-1.2.9/
./configure
make
sudo make install
```

5.3.2 Instalación del Wireshark con Gestores de paquetes

Si usas Debian o sus derivados como Ubuntu, tan solo debes hacer lo siguiente:

```
sudoaptitudeinstallwireshark
```

Para distribuciones básadas en RetHat, como CentOS, hay que agregar una línea de comando más, ya que por defecto no se instala la interfaz gráfica y en lugar de "aptitude" usamos el gestor de paquetes "yum":

```
yum install wireshark

yum install wireshark-gnome
```

5.4 ENTENDIENDO LA INTERFAZ GRÁFICA DE WIRESHARK

Luego de la instalación podrás iniciar el programa con el comando Wireshark o desde el menú de aplicaciones.

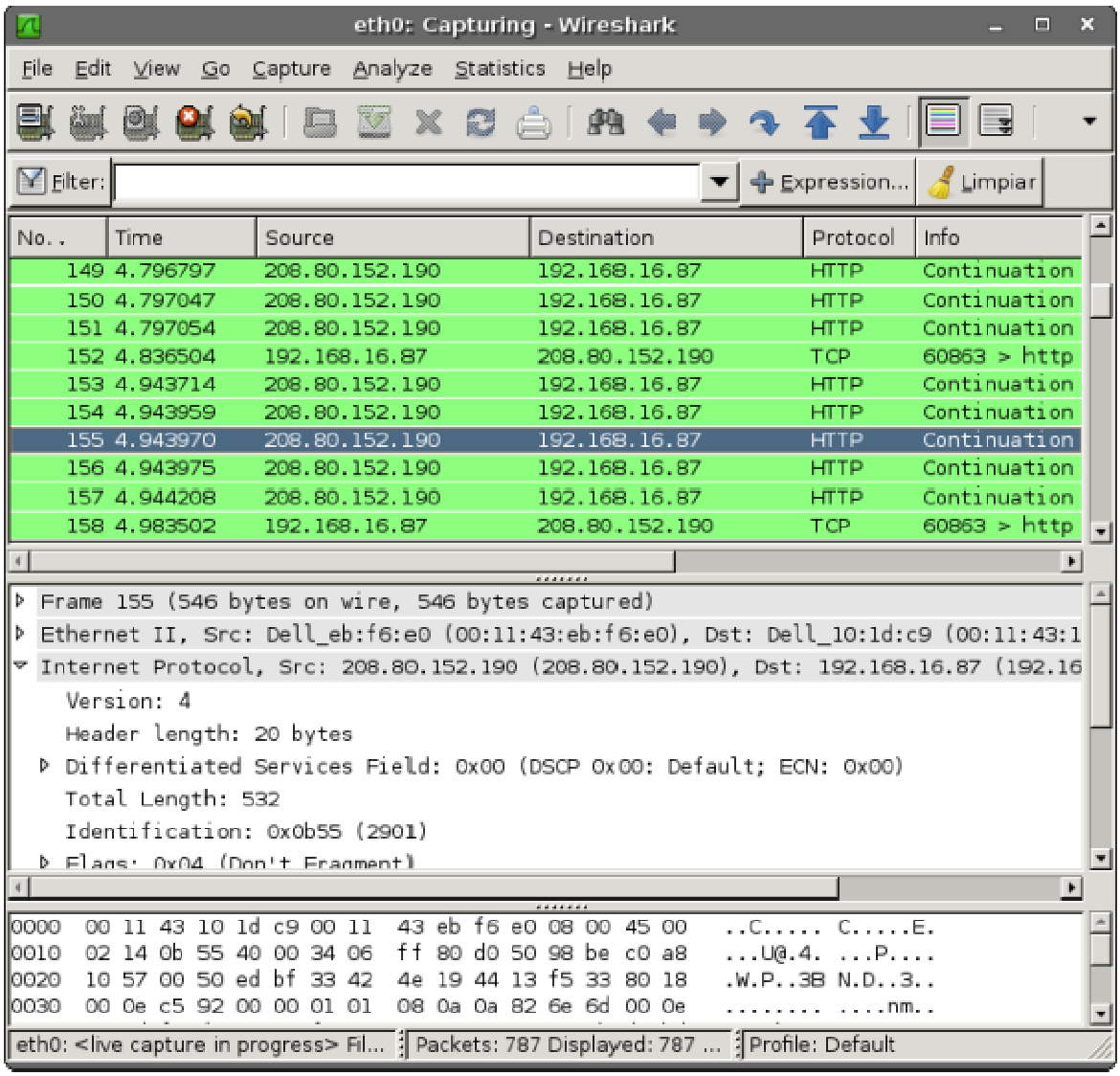


Figura 5.1: Pantalla Principal de Wireshark

La interfaz gráfica de Wireshark (Figura 1.A) está principalmente dividida en las siguientes secciones (de arriba a abajo):

- ✓ La **barra de herramientas**, donde tienes todas las opciones a realizar sobre la pre y pos captura.

- ✓ La **barra de herramientas principal**, donde tienes las *opciones más usadas* en Wireshark.
- ✓ La **barra de filtros**, donde podrás aplicar filtros a la captura actual de manera rápida
- ✓ El **listado de paquetes**, que muestra un resumen de cada paquete que es capturado por Wireshark
- ✓ El **panel de detalles de paquetes** que, una vez seleccionado un paquete en el listado de paquetes, muestra información detallada del mismo
- ✓ El **panel de bytes de paquetes**, que muestra los bytes del paquete seleccionado, y resalta los bytes correspondientes al campo seleccionado en el panel de detalles de paquetes.
- ✓ La **barra de estado**, que muestra algo de información acerca del estado actual de Wireshark y la captura.

ANEXO 6 | EXPERIMENTO 1 - CAPTURA DE LLAMADAS

UTILIZANDO IPV4

La Figura 6.2 muestra el tráfico RTP generado en la captura a través de la herramienta IO Graphs (Ver Figura 6.1) del Wiresahrk, ya que solo nos interesa el tráfico RTP hemos aplicado un filtro. Se puede ver que se produce un mayor tráfico entre los intervalos de 220 - 320 segundos y 850 – 1040 segundos de captura, esto nos indica que se realizaron 2 llamadas que generaron dicho tráfico en ese instante de tiempo, también podemos definir el ancho de banda del mencionado tráfico que sería de alrededor de 46 Kb/s.

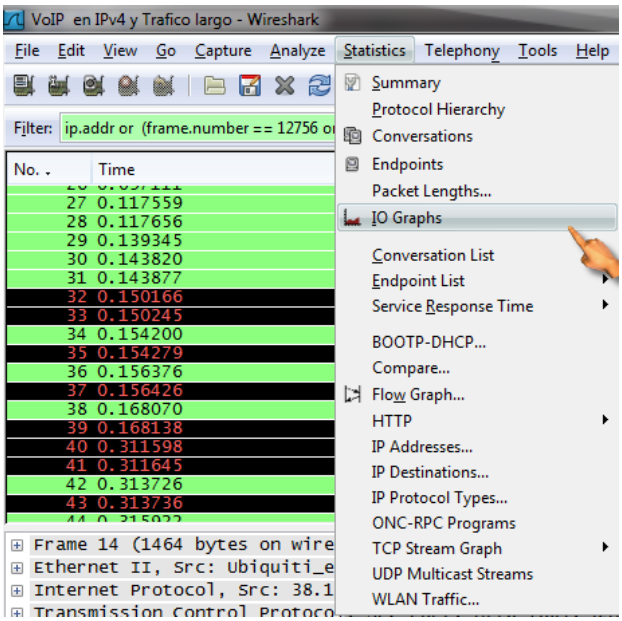


Figura 6.1: IO Graphs

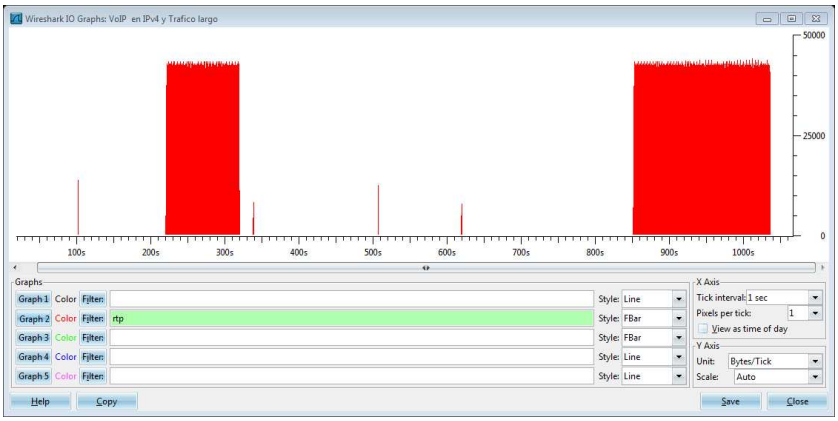


Figura 6.2: Captura de Tráfico RTP

Con esta información podemos utilizar la herramienta VoIPCalls (ver Figura 6.3) del Wireshark para obtener una información más específica de cada llamada. En la Figura 6.4 se muestra una lista de llamadas detectadas en la captura, incluidas las inválidas o de prueba. Según el tráfico analizado las llamadas que generaron tráfico son las que se encuentran seleccionadas en la gráfica.

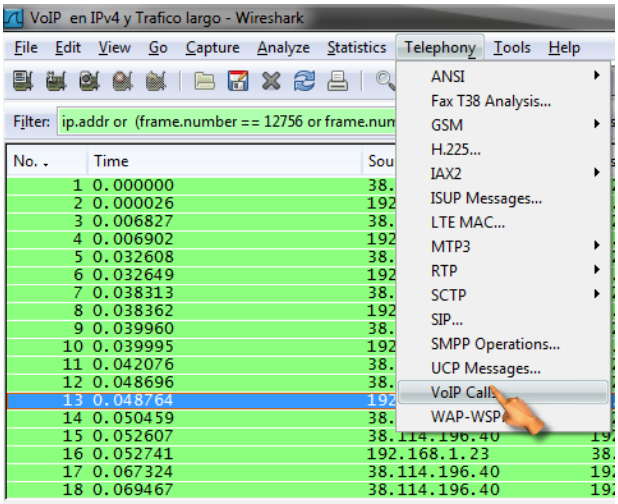


Figura 6.3: VoIPCalls

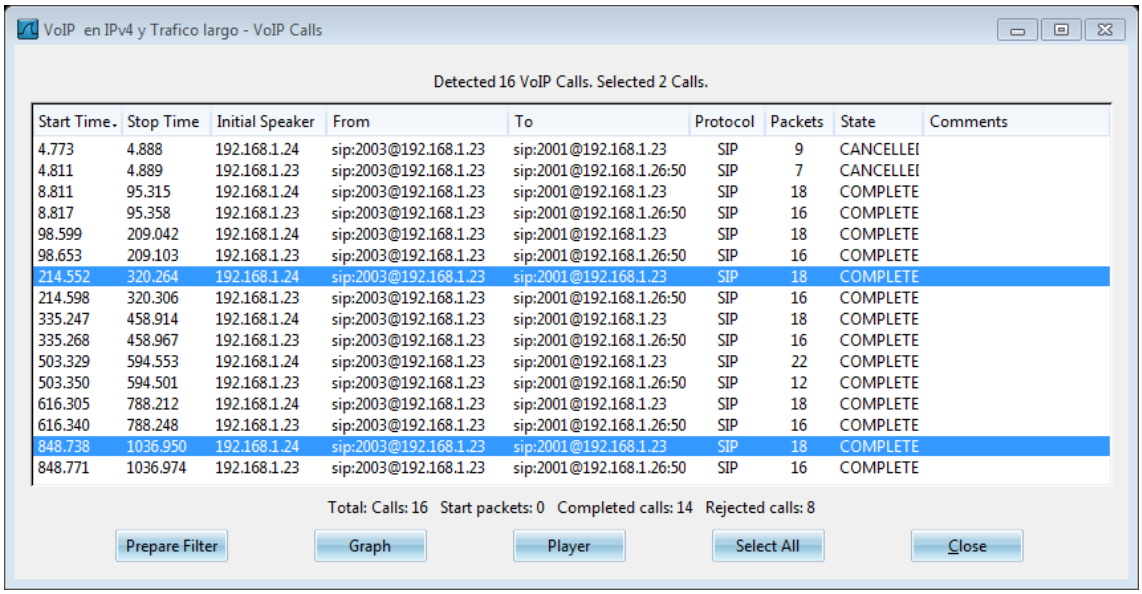


Figura 6.4: Lista de llamadas IPv4 detectadas en la captura

Para nuestro caso vamos a analizar la llamada más extensa que es la segunda, presionando el botón Graph se puede ver en detalle de la comunicación entre el servidor y los clientes endicha llamada (Ver **¡Error! No se encuentra el origen de la referencia.**), la llamada se produce desde la dirección 192.168.1.24 hacia la 192.168.1.26 y todo el tráfico es controlado por el servidor 192.168.1.23.

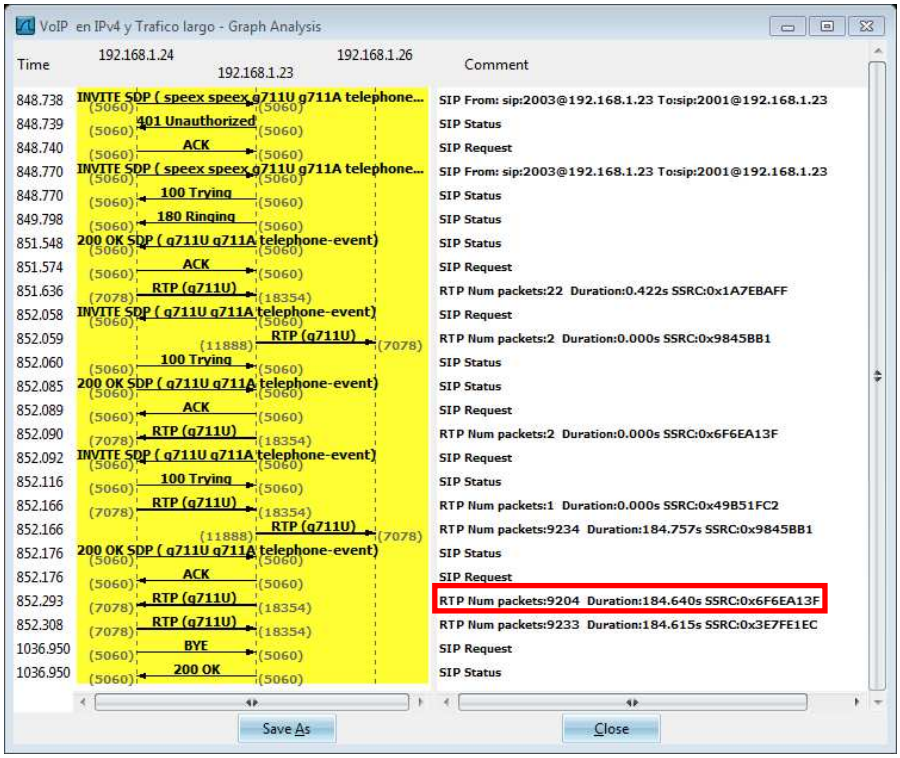


Figura 6.5: Detalle de la llamada

Ahora vamos a hacer uso de otra herramienta de análisis de Wireshark, desde el menú Telephony> RTP > Show Allstreams, que muestra todo el flujo de datos RTP y nos permitirá determinar variables más específicas de comunicación como el jitter, el retardo y los paquetes perdidos. En la Figura 6.6 se puede observar la lista de streams detectados en la captura y vamos a seleccionar el stream que según la información de la llamada contiene el mayor número de paquetes es decir donde hay más probabilidad de retardos y problemas de comunicación. Luego de seleccionar el stream simplemente damos click en Analize y nos aparecerán los paquetes que contiene el stream y principalmente datos específicos de la comunicación (verFigura 6.7)

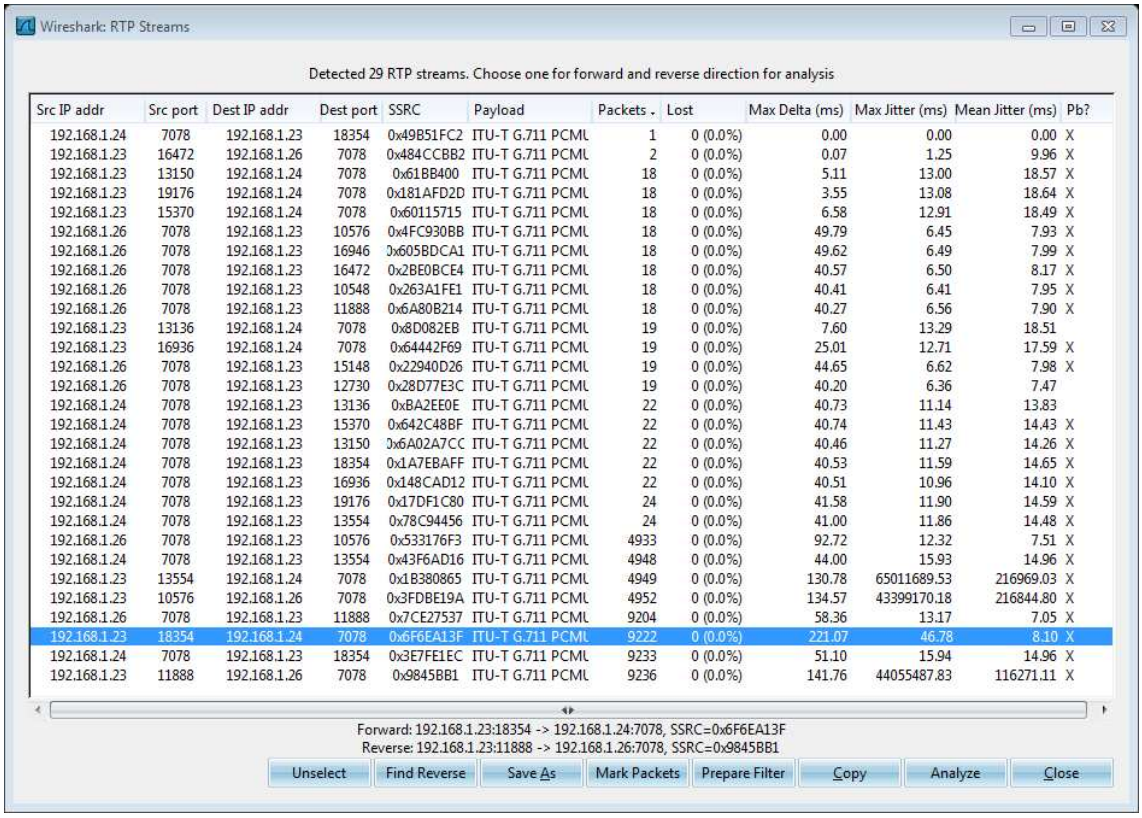


Figura 6.6: RTP Streams

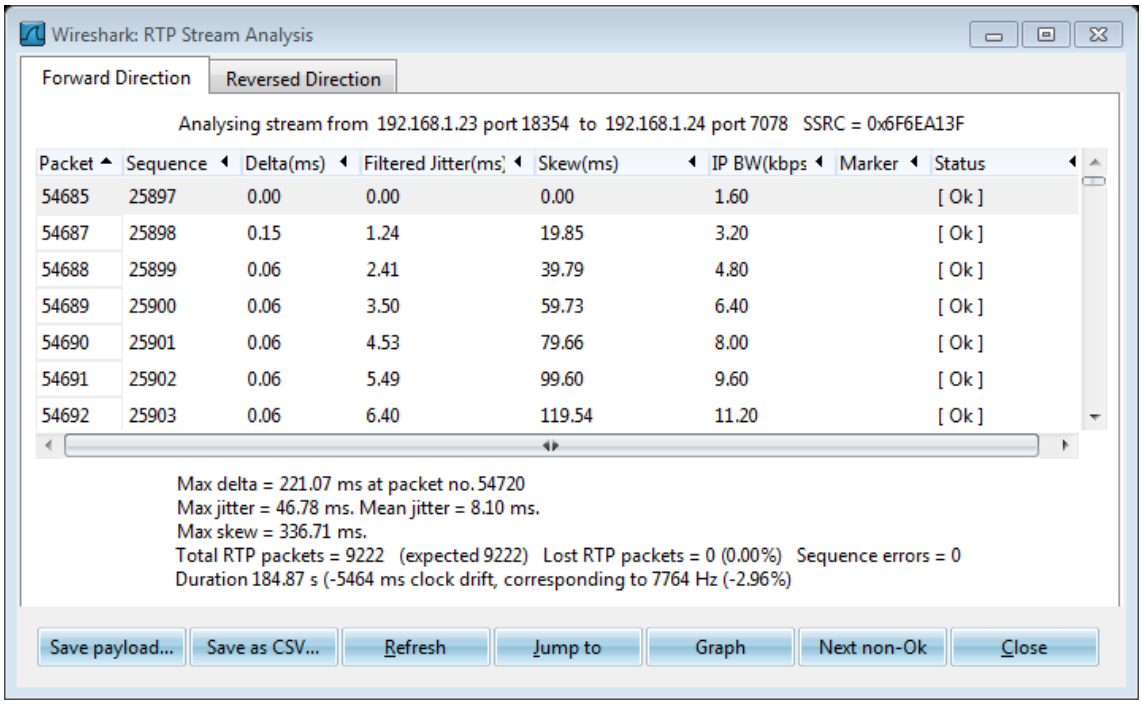


Figura 6.7: Analisis del stream seleccionado

Como resultado tenemos valores promedio de las principales variables que intervienen en la comunicación, los mismos que se muestran en la Tabla 6.1

Pérdida de Paquetes	0 %
Jitter Máximo	46.78
Jitter Promedio	8.10
Delta Máximo	221.07

Tabla 6.1: Resultados Captura IPv4

Por último se va a probar la calidad de Audio con una de las herramientas más interesantes con las que cuenta Wireshrark, con la cual podemos escuchar el audio capturado de la llamada e ir variando el "buffer jitter". Ha esta Herramienta podemos acceder desde la lista de llamadas VoIP (Figura 6.4), seleccionando la llamada y presionando Player.

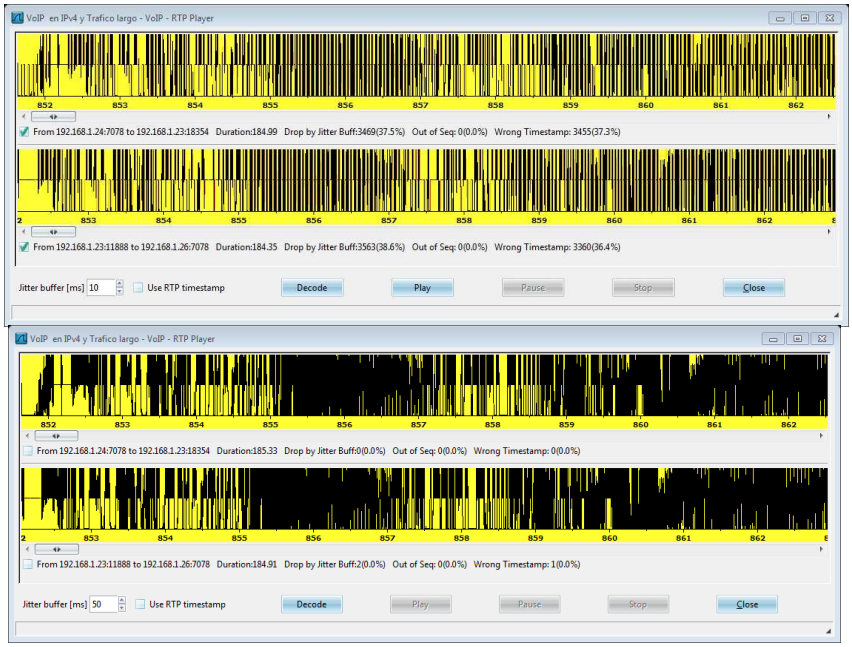


Figura 6.8: Capturas de Audio con diferentes jitter buffer

En la Figura 6.8 se muestran las capturas de audio de la llamada con valores de jitter buffer de 10 y 50 respectivamente.

ANEXO 7 | EXPERIMENTO 2 - CAPTURA DE LLAMADAS

UTILIZANDO IPV6

En la Figura 7.1 se puede ver que se produce un mayor tráfico en el intervalo de 90 - 165 segundos de captura, esto nos indica que se realizó una llamada continua en este tiempo, también podemos definir el ancho de banda que sería de alrededor de 48 Kb/s.

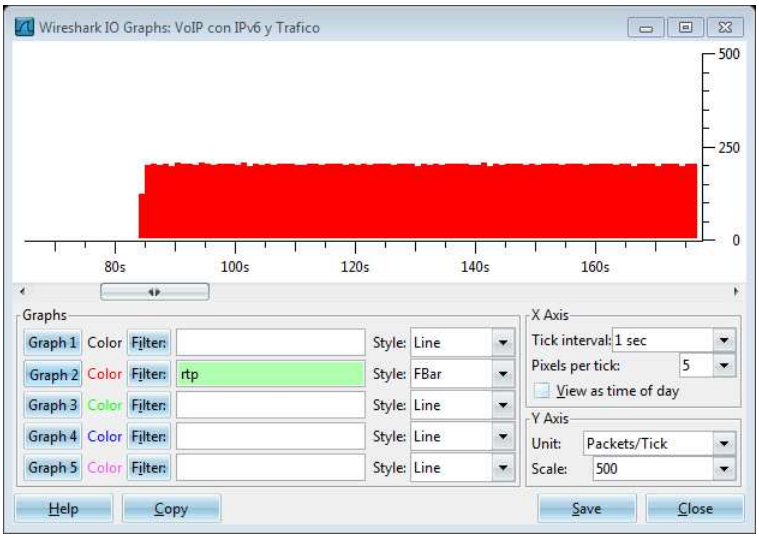


Figura 7.1: Captura Trafico RTP

Con esta información podemos utilizar la herramienta VoIPCalls para obtener una información más específica de cada llamada. En la Figura 7.2se muestra una lista de llamadas detectadas en la captura.

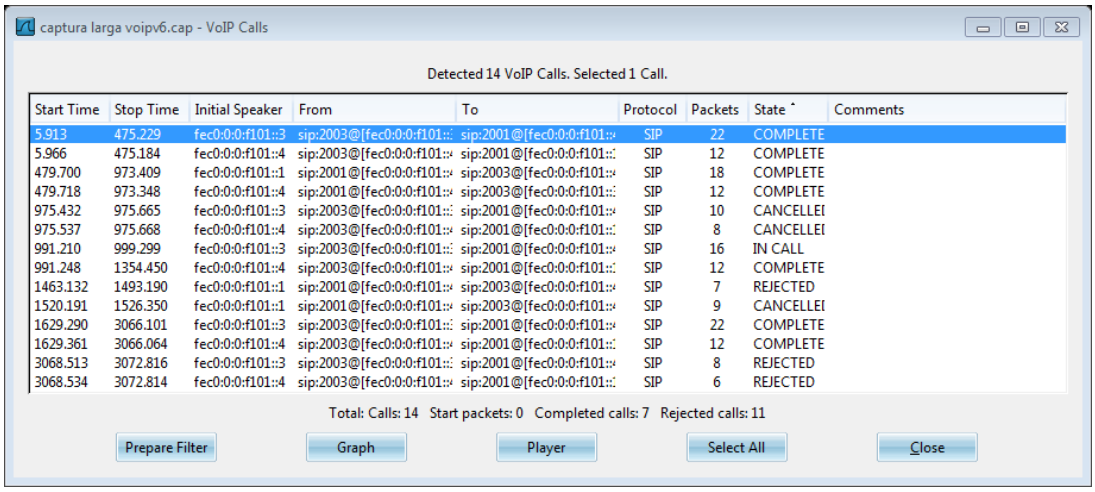


Figura 7.2: Lista de Llamadas IPv6

Seleccionando la llamada y presionando el botón Graph se puede ver en detalle la comunicación entre el servidor y los clientes (Ver Figura 7.3), la llamada se produce desde la dirección fec0:0:0:f1:01::3 hacia la fec0:0:0:f1:01::1 y todo el tráfico es controlado por el servidor fec0:0:0:f1:01::4.

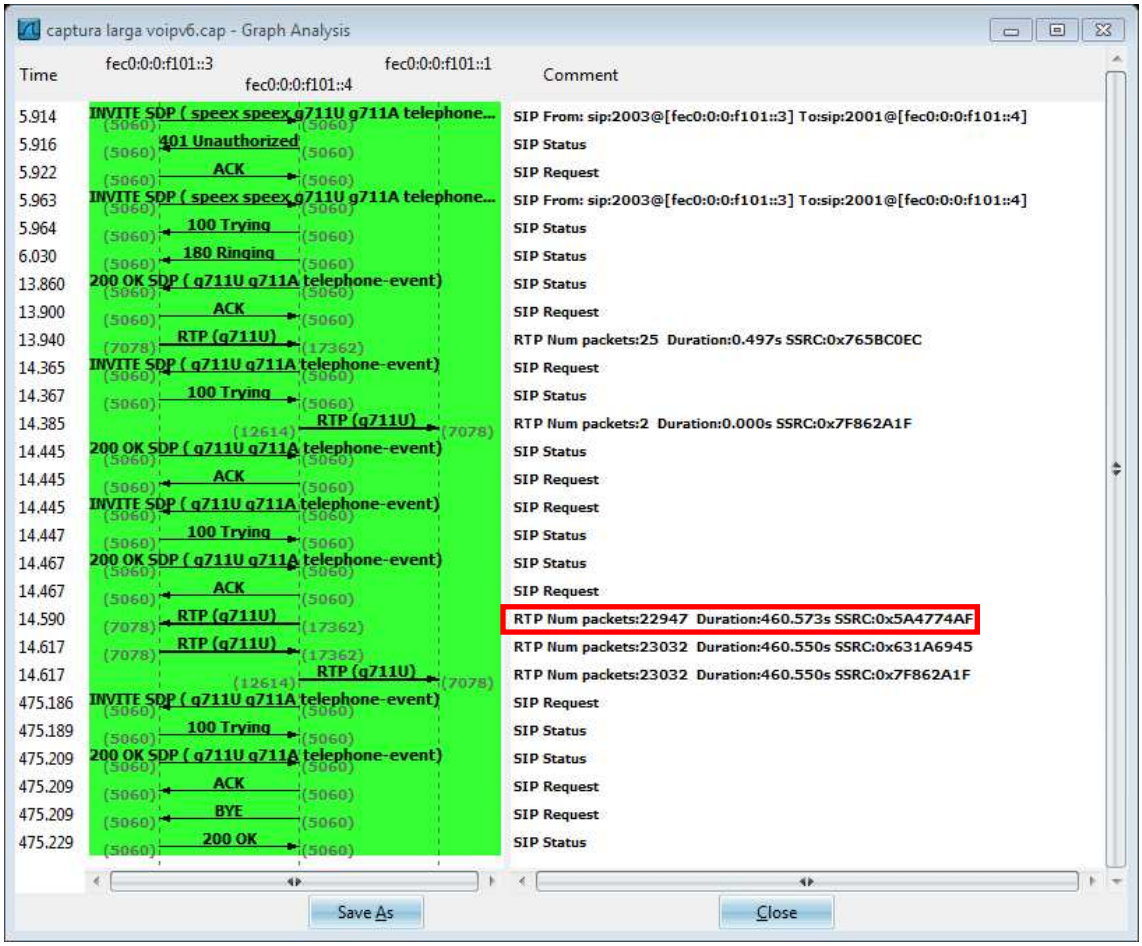


Figura 7.3: Detalle de la llamada

En la Figura 7.4 se puede observar la lista de streams detectados en la captura e igual que en experimento anterior vamos a seleccionar el stream que según la información de la llamada contiene el mayor número de paquetes. Luego

de seleccionar el stream simplemente damos click en Analize y nos aparecerán los paquetes que contiene el stream y principalmente datos específicos de la comunicación (verFigura 7.5)

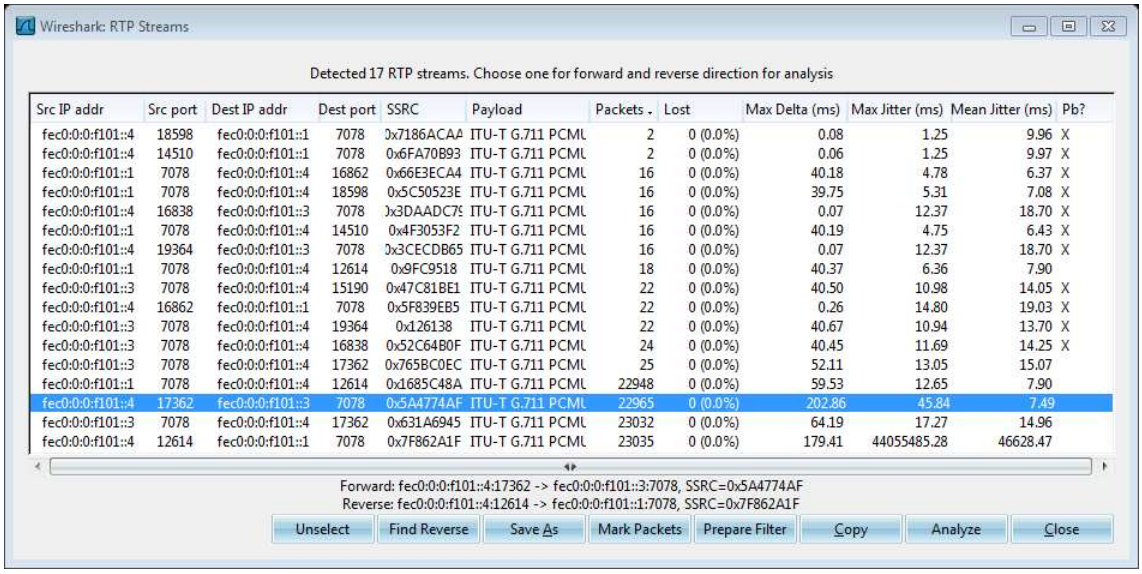


Figura 7.4: RTP Streams

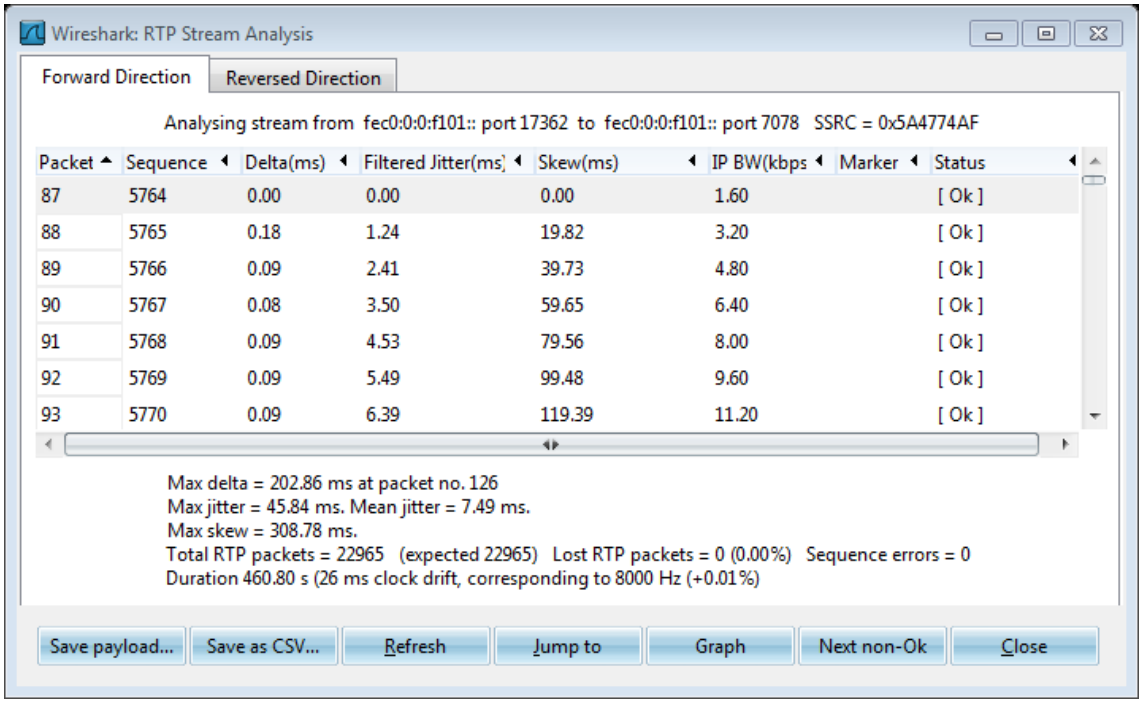


Figura 7.5: RTP StreamAnalysis

Como resultado tenemos valores promedio de las principales variables que intervienen en la comunicación, los mismos que se muestran en la Tabla 7.1

Pérdida de Paquetes	0 %
Jitter Máximo	45.84
Jitter Promedio	7.49
Delta Máximo	202.86

Tabla 7.1: Resultados Captura IPv6

En la Figura 7.6 se muestran las capturas de audio de la llamada con valores de jitter buffer de 10 y 50 respectivamente.

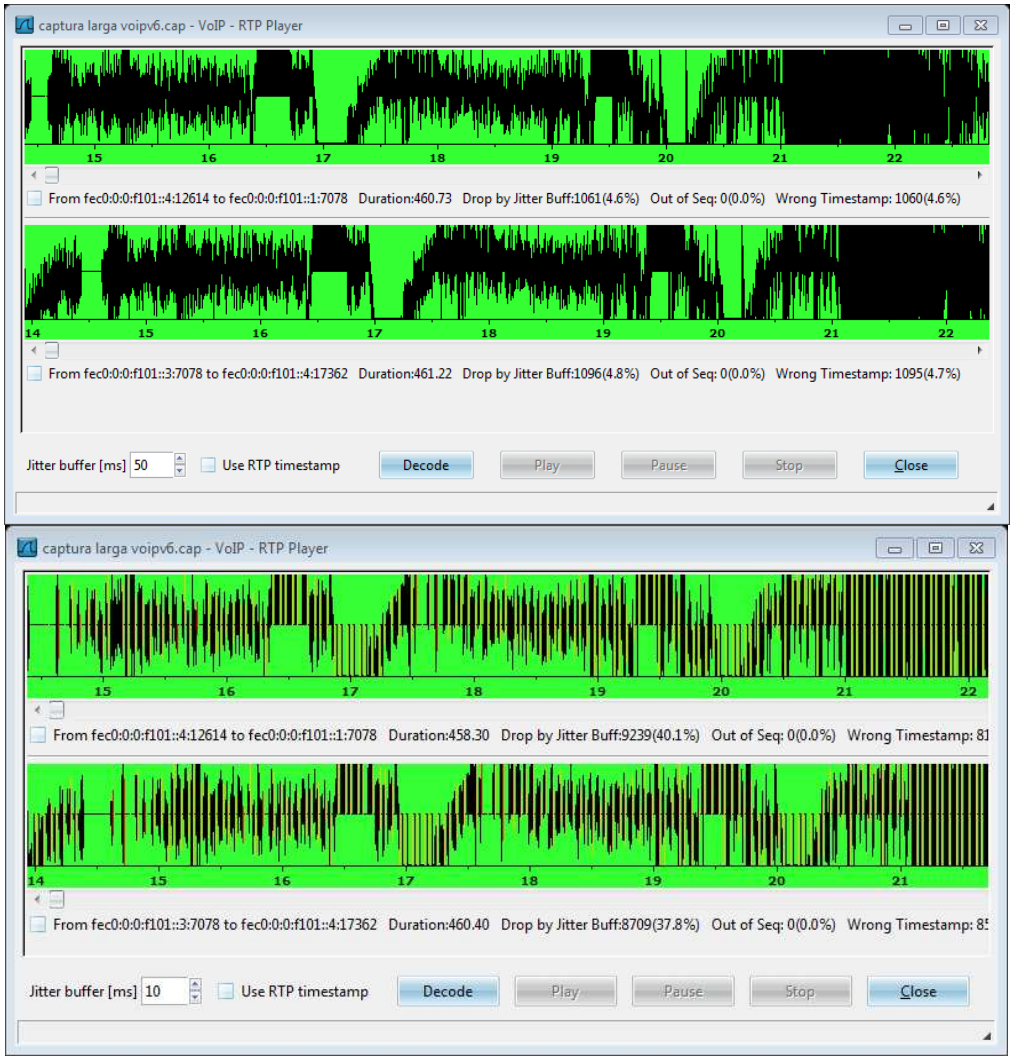


Figura 7.6: Capturas de Audio llamada IPv6

ANEXO 8 | ENCUESTA

ESCUELA SUPERIOR POLITÉCNICA DEL CHIMBORAZO

ENCUESTA

OBJETIVO:

El objetivo dela presente Encuesta es analizar el funcionamiento de dos sistemas de Voz sobre IP y determinar cuál se implementará a futuro

INSTRUCTIVO:

- ✓ Lea cuidadosamente las preguntas
- ✓ Marque con una X o llenando el casillero de la opción que corresponda

CUESTIONARIO:

- ✓ ¿Usaría o recomendaría el uso de estos sistemas telefónicos en su hogar o lugar de trabajo?

✓ Si	☐	✓ No	☐	✓ Tal vez	☐

- ✓ ¿En cuál de los dos sistemas se escuchó más clara la llamada?

✓ Nº 1	☐	✓ Nº 2	☐	✓ Igual	☐

✓ ✓ ✓ ✓ ✓ ✓

✓ ¿Cuál de los dos sistemas le pareció más fácil de utilizar?

✓ ✓ ✓ ✓ ✓ ✓

✓ Nº 1 ✓ ✓ Nº 2 ✓ ✓ Igual ✓

✓ ✓ ✓ ✓ ✓ ✓

GRACIAS POR SU COLABORACIÓN!!!

ANEXO 9 | TABLA DE DISTRIBUCIÓN DE X^2

TABLA DE DISTRIBUCIÓN DE X^2

	Probabilidad de un valor superior - *Alfa* (α)				
Grados libertad	0,1	0,05	0,025	0,01	0,005
1	2,71	3,84	5,02	6,63	7,88
2	4,61	5,99	7,38	9,21	10,60
3	6,25	7,81	9,35	11,34	12,84
4	7,78	9,49	11,14	13,28	14,86
5	9,24	11,07	12,83	15,09	16,75
6	10,64	12,59	14,45	16,81	18,55
7	12,02	14,07	16,01	18,48	20,28
8	13,36	15,51	17,53	20,09	21,95
9	14,68	16,92	19,02	21,67	23,59
10	15,99	18,31	20,48	23,21	25,19
11	17,28	19,68	21,92	24,73	26,76
12	18,55	21,03	23,34	26,22	28,30
13	19,81	22,36	24,74	27,69	29,82
14	21,06	23,68	26,12	29,14	31,32
15	22,31	25,00	27,49	30,58	32,80
16	23,54	26,30	28,85	32,00	34,27
17	24,77	27,59	30,19	33,41	35,72
18	25,99	28,87	31,53	34,81	37,16
19	27,20	30,14	32,85	36,19	38,58
20	28,41	31,41	34,17	37,57	40,00
21	29,62	32,67	35,48	38,93	41,40
22	30,81	33,92	36,78	40,29	42,80
23	32,01	35,17	38,08	41,64	44,18
24	33,20	36,42	39,36	42,98	45,56
25	34,38	37,65	40,65	44,31	46,93
26	35,56	38,89	41,92	45,64	48,29
27	36,74	40,11	43,19	46,96	49,65
28	37,92	41,34	44,46	48,28	50,99
29	39,09	42,56	45,72	49,59	52,34
30	40,26	43,77	46,98	50,89	53,67
40	51,81	55,76	59,34	63,69	66,77
50	63,17	67,50	71,42	76,15	79,49
60	74,40	79,08	83,30	88,38	91,95
70	85,53	90,53	95,02	100,43	104,21
80	96,58	101,88	106,63	112,33	116,32
90	107,57	113,15	118,14	124,12	128,30
100	118,50	124,34	129,56	135,81	140,17