



**INSTITUTO POLITÉCNICO NACIONAL**

---

**ESCUELA SUPERIOR DE INGENIERÍA MECÁNICA Y ELÉCTRICA  
SECCIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN  
ESIME-ZACATENCO**

**IMPLEMENTACIÓN DE UN SISTEMA  
DE VOZ IP CON DISPOSITIVOS  
MÓVILES**

**T E S I S**

QUE PARA OBTENER EL GRADO DE  
**MAESTRO EN CIENCIAS EN INGENIERÍA DE TELECOMUNICACIONES**

PRESENTA

**ING. MIGUEL ANGEL DANIEL FLORES**

DIRECTOR DE TESIS

**M.C. MARCO ANTONIO ACEVEDO MOSQUEDA.**



México, D.F.

2013



**INSTITUTO POLITÉCNICO NACIONAL**  
**SECRETARÍA DE INVESTIGACIÓN Y POSGRADO**  
**ACTA DE REVISIÓN DE TESIS**

En la Ciudad de México, D. F. siendo las 8:30 horas del día 28 del mes de Junio del 2013 se reunieron los miembros de la Comisión Revisora de la Tesis, designada por el Colegio de Profesores de Estudios de Posgrado e Investigación de ESIME-Zacatenco para examinar la tesis titulada:

**“IMPLEMENTACIÓN DE UN SISTEMA DE VOZ IP CON DISPOSITIVOS MÓVILES”**

Presentada por el alumno:

<b>DANIEL</b> Apellido paterno	<b>FLORES</b> Apellido materno	<b>MIGUEL ANGEL</b> Nombre(s)							
		Con registro:	A	1	1	0	8	9	6

aspirante de:

**MAESTRO EN CIENCIAS EN INGENIERÍA DE TELECOMUNICACIONES**

Después de intercambiar opiniones, los miembros de la Comisión manifestaron **APROBAR LA TESIS**, en virtud de que satisface los requisitos señalados por las disposiciones reglamentarias vigentes.

LA COMISIÓN REVISORA

Director(a) de tesis

**M. EN C MARCO ANTONIO ACEVEDO MOSQUEDA**

PRESIDENTE

**DR. JORGE ROBERTO SOSA PEDROZA**

SEGUNDO VOCAL

**DRA. MARIA ELENA ACEVEDO MOSQUEDA**

TERCER VOCAL

**M. EN C. CARLOS RIOS RIOS**

SECRETARIO

**M. EN C. MIGUEL SÁNCHEZ MERAZ**

PRESIDENTE DEL COLEGIO DE PROFESORES

**DR. MAURO ALBERTO ENCISO AGUILAR**






**INSTITUTO POLITÉCNICO NACIONAL**  
**SECRETARÍA DE INVESTIGACIÓN Y POSGRADO**

*CARTA CESIÓN DE DERECHOS*

En la Ciudad de México, D.F. el día 1 del mes de julio del año 2013, el que suscribe **MIGUEL ANGEL DANIEL FLORES** alumno del Programa de **MAESTRÍA EN CIENCIAS EN INGENIERÍA DE TELECOMUNICACIONES**, con número de registro **A110896**, adscrito a la Sección de Estudios de Posgrado e Investigación de la ESIME Unidad Zacatenco, manifiesto que es el autor intelectual del presente trabajo de Tesis bajo la dirección del **M.C. MARCO ANTONIO ACEVEDO MOSQUEDA** y cede los derechos del trabajo titulado “**IMPLEMENTACIÓN DE UN SISTEMA DE VOZ IP CON DISPOSITIVOS MÓVILES**”, al Instituto Politécnico Nacional para su difusión, con fines académicos y de investigación.

Los usuarios de la información no deben reproducir el contenido textual, gráficas o datos del trabajo sin el permiso expreso del autor y/o director del trabajo. Este puede ser obtenido escribiendo a la siguiente dirección: [madanielf@yahoo.es](mailto:madanielf@yahoo.es).

Si el permiso se otorga, el usuario deberá dar el agradecimiento correspondiente y citar la fuente del mismo.



---

MIGUEL ANGEL DANIEL FLORES



## **RESUMEN.**

En la actualidad existe un gran auge de la telefonía empleando protocolos de Internet, esta tecnología es conocida como Telefonía IP y usa estos protocolos para el establecimiento de comunicaciones de voz.

De igual manera existe un crecimiento en el desarrollo de dispositivos móviles con gran capacidad de procesamiento y almacenamiento para cubrir las demandas de las tecnologías actuales.

De manera paralela al crecimiento de los dispositivos móviles se ha generado una gran demanda de aplicaciones, estas van desde aplicaciones de entretenimiento, sistemas de posicionamiento global, fotografía, video y por supuesto aplicaciones de voz IP.

Para la implementación de un conmutador telefónico existen varias soluciones, desde Call Manager de Cisco con las ventajas y desventajas que con lleva el equipamiento, licenciamiento, actualización y mantenimiento de este tipo de sistemas, hasta centrales telefónicas basadas en software libre como Asterisk y Freeswitch.

En el presente trabajo de tesis se implementó un sistema de voz IP a partir de una central telefónica de código libre (*Asterisk*) con dispositivos móviles de los sistemas operativos de mayor presencia a nivel mundial, empleando aplicaciones gratuitas de voz IP existentes en el mercado.

## **ABSTRACT.**

Currently there is a boom in telephony using Internet protocols, this technology is known as IP Telephony and uses these protocols to establish voice communications.

Likewise, there is a growth in the development of mobile devices with high processing capacity and storage to meet the demands of today's technologies.

Parallel to the growth of mobile devices has created a high demand for applications, these applications range from entertainment, global positioning systems, photography, video and of course IP voice applications.

For the implementation of a PBX there are many solutions, from Cisco Call Manager with the advantages and disadvantages that carries the equipment, licensing, updating and maintenance of such systems to PBX based on free software like Asterisk and Freeswitch.

In this thesis we implemented a VoIP system from an open source PBX (Asterisk) with mobile devices operating systems greater global presence, using free IP voice applications on the market.

Resumen

Abstract

Índice de tablas y figuras.

Descripción del problema

Justificación

Objetivo

Modelo propuesto

## **Capítulo 1: Introducción**

## **Capítulo 2: Marco teórico**

- 2.1 La telefonía.
- 2.2 La telefonía móvil.
- 2.3 Internet.
  - 2.3.1 Protocolo IP.
  - 2.3.2 Protocolo TCP.
  - 2.3.3 Protocolo UDP.
- 2.4 Introducción a la Voz IP (VoIP).
  - 2.4.1 Proceso de digitalización.
    - 2.4.1.1 Filtrado
    - 2.4.1.2 Muestreo y retención.
    - 2.4.1.3 Cuantificación.
    - 2.4.1.4 Codificación.
  - 2.4.2 Telefonía conmutada por circuitos.
  - 2.4.3 Telefonía conmutada por paquetes.
  - 2.4.4 Protocolos.
    - 2.4.4.1 Inicio de Sesión.
    - 2.4.4.2 Audio.
- 2.5 Central Telefónica Asterisk.
- 2.6 Aplicaciones de VoIP para dispositivos móviles.

## **Capítulo 3: Modelo Propuesto.**

- 3.1 Descripción del modelo.
- 3.2 Servidor de voz.
  - 3.2.1 Instalación del sistema operativo.
  - 3.2.2 Instalación y configuración de la central telefónica.
    - 3.2.2.1 Configuración del sistema.
    - 3.2.2.2 Instalación de la central telefónica.
    - 3.2.2.3 Instalación del codec AMR-NB.

3.3 Análisis de aplicaciones de voz IP.

3.3.1 Vmobile.

3.3.2 Join Express.

3.3.3 Linphone.

**Capítulo 4: Resultados.**

**Capítulo 5: Conclusiones y Trabajos Futuros.**

## ÍNDICE DE TABLAS Y FIGURAS.

### ÍNDICE DE TABLAS.

Tabla 2.1	Codecs más usados para telefonía.
Tabla 2.2	Peticiones del protocolo SIP.
Tabla 2.3	Respuestas del protocolo SIP.
Tabla 4.1	Dispositivos móviles para la realización de pruebas.
Tabla 4.2	Resumen de pruebas entre dispositivos móviles

### ÍNDICE DE FIGURAS.

Figura 2.1	Diagrama de celdas en telefonía celular.
Figura 2.2	Arquitectura TCP/IP.
Figura 2.3	Encapsulamiento de paquete IP.
Figura 2.4	Estructura de paquete IP.
Figura 2.5	Formato de paquete TCP.
Figura 2.6	Formato de paquete UDP.
Figura 2.7	Arquitectura de Voz IP
Figura 2-8	Esquema de protocolo SIP.
Figura 2.9	Proceso de establecimiento de llamada entre dos usuarios.
Figura 2.10	Diseño modular de Asterisk
Figura 2.11	Aplicaciones de voz IP para dispositivos móviles.
Figura 3.1	Diagrama del modelo propuesto.
Figura 3.2	Inicio de instalación de CentOS 6.
Figura 3.3	Comprobación del disco duro en CentOS 6.
Figura 3.4	Inicio de configuración de CentOS 6.
Figura 3.5	Advertencia del dispositivo de almacenamiento en CentOS 6.
Figura 3.6	Asignación del host en CentOS 6.
Figura 3.7	Selección de zona horaria y ubicación en CentOS 6.
Figura 3.8	Selección de contraseña de root en CentOS 6.
Figura 3.9	Selección del tipo de instalación en CentOS 6.



- Figura 3.10 Instalación de CentOS 6.
- Figura 3.11 Finalización de la instalación de CentOS 6.
- Figura 3.12 Registro para inicio de sesión en modo consola para CentOS 6.
- Figura 3.13 Codecs instalados en la central telefónica.
- Figura 3.14 Tiempo para la realización de Transcoding en asterisk 1.8.16.0.
- Figura 4.1 Registro de usuarios.
- Figura 4.2 Usuarios registrador en el servidor.
- Figura 4.3 Características de un usuario específico.
- Figura 4.4 Instrucción sip set debug on.
- Figura 4.5 Proceso de una llamada.
- Figura 4.6 Flujo de paquetes de voz al realizar una llamada.
- Figura 4.7 Interfaz gráfica del analizador de protocolos Wireshark.
- Figura 4.8 Filtrado de protocolo SIP en Wireshark.
- Figura 4.9 Flujo del protocolo SIP para el establecimiento de una sesión en Wireshark.
- Figura 4.10 Captura de paquetes RTP en Wireshark.
- Figura 4.11 Decodificación de una llamada en Wireshark.
- Figura 4.12 Llamada entre Blackberry e iPhone codec PCM ley A.
- Figura 4.13 Llamada entre Blackberry e iPhone codec PCM ley  $\mu$
- Figura 4.14 Llamada entre Blackberry codec AMR.
- Figura 4.15 Decodificación de llamada entre Blackberry codec AMR.

## **DESCRIPCIÓN DEL PROBLEMA.**

Actualmente en el mercado existen aplicaciones para dispositivos móviles que realizan comunicaciones de voz IP, estas aplicaciones se encuentran enfocadas principalmente a dispositivos móviles con sistemas operativos Android e iOS, debido a las facilidades que proporcionan estos sistemas operativos, inclusive cuentan con aplicaciones nativas desarrolladas por las empresas que fabrican dichos dispositivos.

Por lo que respecta a otra de las grandes compañías distribuidoras de teléfonos inteligentes como Blackberry, la cual ha tenido una gran penetración en el mercado de la telefonía celular principalmente enfocada a los campos empresarial e institucional, no cuenta con una aplicación de voz IP nativa desarrollada por citada empresa, por lo que algunas compañías de desarrollo de software y proveedores de voz IP se han dado a la tarea de desarrollar aplicaciones de voz IP para estos dispositivos móviles.

## **JUSTIFICACIÓN.**

En la actualidad existe una marcada tendencia a la adopción de la tecnología de voz IP para las comunicaciones de voz.

La telefonía móvil, no se puede quedar al margen de este desarrollo tecnología, es por ello que algunas de las empresas mundiales distribuidoras de dispositivos móviles ofrecen esta nueva tecnología en sus equipos.

Por lo que se refiere a los dispositivos móviles Blackberry, no cuentan con una aplicación nativa para su sistema operativo.

Se considera interesante realizar una comparación entre las aplicaciones de voz IP disponibles en el mercado para estos dispositivos móviles y presentar una solución actual a esta problemática.

Por lo anterior es necesario realizar pruebas con las aplicaciones de voz IP disponibles en el mercado, compatibles con teléfonos inteligentes de la marca de Blackberry y con centrales telefónicas basadas en software de código libre con licencia GNU como asterisk en entornos controlados como redes privadas, con el fin de verificar cual o cuales de estas aplicaciones están en posibilidad de proporcionar servicio de voz con dispositivos móviles, con las ventajas y desventajas inherentes a esta tecnología aún en expansión

## **OBJETIVO.**

Implementar un sistema de voz IP empleando dispositivos móviles.

### **Objetivos particulares.**

Investigar e implementar una central telefónica basada en software.

Investigar, implementar y analizar aplicaciones de voz IP para dispositivos móviles.

## Capítulo 1

### **INTRODUCCIÓN.**

La voz es la principal forma comunicarse de los seres humanos, debido a esto ha sido necesaria la investigación y desarrollo de tecnologías que nos brinden este servicio.

El desarrollo tecnológico en esta materia inicia con la invención del teléfono, el cual nos brinda la posibilidad de establecer comunicación de voz entres dos o más sitios remotos, en un inicio este revolucionario invento casi no fue explotado, pero en la actualidad es un servicio necesario.

La continua evolución de las tecnologías ha impuesto grandes retos en el campo de las comunicaciones, lo que hace que la investigación en este rubro de la ciencia tenga que adaptarse a las nuevas tecnologías en campos afines para poder ser explotadas y aprovechadas por las comunicaciones.

A la par con el crecimiento de las redes de cómputo se ha creado un desarrollo simbiótico en las comunicaciones de voz empleando protocolos propios de redes de cómputo, las cuales se han expandido a nivel mundial pudiendo alcanzar a un usuario ubicado en cualquier punto del globo terráqueo que cuente con una conexión a internet, creando una convergencia de servicios y aplicaciones.

Debido a la evolución y expansión de las redes de cómputo y principalmente de internet, actualmente existe una gran migración de comunicaciones de voz de la telefonía convencional (conmutación de circuitos) a la telefonía IP (conmutación de paquetes) conocida mundialmente como voz IP.

La voz IP trae consigo muchos beneficios, uno de los principales es el ahorro en llamadas telefónicas empleando los enlaces tradicionales de voz que proporciona la telefonía convencional, principalmente en las llamadas de larga distancia, por otro lado la implementación de esta tecnología permite la convergencia de voz y datos en una sola red, por lo que únicamente es necesario el empleo de cableado de red para ambos servicios.

Otra de las ventajas de la voz IP es el empleo eficiente del ancho de banda del canal, ya que en la telefonía tradicional cada vez que se genera una llamada se reserva un canal dedicado para el establecimiento de la comunicación de voz generando con ello un dispendio de ancho de banda debido a que por el canal solo se establece una llamada a la vez. En la tecnología de voz IP no sucede lo mismo ya que al establecer una llamada no se reserva un canal dedicado para ello, sino que las señales de voz son procesadas, empaquetadas y transmitidas por redes de cómputo y no necesariamente siguen el mismo camino, si no que pueden ser transmitidas por caminos distintos hasta llegar a su destino final y una vez ahí se realiza el proceso inverso para la obtención de las señales de voz, pudiendo establecer varias llamadas empleando el mismo canal haciendo un uso más eficiente del ancho de banda.

La voz IP es llamada de esta manera debido a que emplea protocolos de internet (*IP Internet Protocols*) no solo con dispositivos fijos como teléfonos IP o aplicaciones de voz IP conocidos como teléfonos de software o softphones para equipos de cómputo, sino también en dispositivos móviles como teléfonos inteligentes o smartphones e inclusive con equipos de telefonía convencional que cuenten con un Adaptador Telefónico Analógico (*ATA*).

Principalmente a nivel empresarial, corporativo e institucional es necesario proporcionar servicios de voz y datos a su platilla laboral, una solución para satisfacer a estos usuarios sería la contratación de líneas dedicadas de telefonía tradicional con los altos costos que esto conlleva, otra solución sería la instalación de una central telefónica convencional, lo cual también implica un razonable costo inicial, este último caso proporcionaría servicio sin costo entre los usuarios registrados en citada central telefónica y además con la contratación de líneas troncales y un correcto plan de marcación se estaría en posibilidad de proporcionar servicio hacia el exterior de la central telefónica, en ambas soluciones no se han considerado los servicios de datos, por ejemplo intranet de la institución o inclusive internet, que dependiendo el número de usuarios y el cada vez más necesario servicio, generaría un costo extra.

Considerando la problemática anterior, existe una solución más económica y cada vez más eficiente, la instalación de una red de cómputo, la cual nos podría proporcionar servicios de

datos y mediante la instalación de centrales telefónicas de voz IP, el servicio de voz, irremediablemente generaría un costo inicial, el cual no sería tan oneroso como la instalación de una central telefónica convencional, en este último escenario solamente se realizaría una sola instalación, la red de datos, proporcionando ambos servicios a los usuarios.

La voz IP no es ajena a la telefonía móvil, actualmente las empresas proveedoras del servicio de telefonía móvil nos proporcionan servicios de voz y datos, inclusive sin contratar planes de renta, de igual forma, la gran mayoría de dispositivos móviles de tecnología reciente cuentan con la posibilidad de conexión a redes de datos mediante soluciones inalámbricas, haciendo posible la comunicación de voz empleando el canal de datos proporcionado por el proveedor de telefonía celular o inclusive mediante la conexión a internet empleando las soluciones ya mencionadas, lo cual se ve reflejado en los ahorros generados a los usuarios finales y por supuesto al administrador del sistema.

Por otro lado la investigación en el campo de la electrónica ha permitido la creación de dispositivos móviles cada vez más poderosos, permitiendo en la actualidad no solo contar con un dispositivo de radio de dos vías, sino que además contamos con equipos con capacidad de almacenamiento y procesamiento de datos muy poderosos logrando tener en nuestras manos equipos de cómputo portátil con capacidad de conexión a redes de computadoras.

Las redes de telefonía móvil han tenido que evolucionar y adaptarse a los cambios que la tecnología ha marcado, por lo que las actuales redes de telefonía móvil tienen la capacidad de proporcionar servicios de voz y datos, lo cual nos brinda servicios adicionales como navegación por internet y envío de mensajes multimedia entre muchos otros más

La implantación de estas tecnologías ha generado un incipiente desarrollo de aplicaciones para dispositivos móviles enfocados en diferentes campos como el entretenimiento, difusión de noticias y redes sociales sin olvidar el objetivo por lo que fueron creados, las comunicaciones de voz.

Lo anterior no sería posible si no existieron estándares internacionales que regulan y proporcionan estandarización tecnológica, mediante normas y protocolos que son

adoptados por los fabricantes y proveedores de servicio para la convivencia de las tecnologías, las principales organizaciones reguladores en este ámbito son las IETF (*Internet Engineering Task Force*) y la ITU (*International Telecommunication Union*)

La voz IP se encuentra basada en dos tipos de protocolos necesarios para su desempeño, los protocolos de señalización como SIP (*Session Initial Protocol*) y H323, y los protocolos de tiempo real como RTP (*Real-time Transport Protocol*), los cuales trataremos más ampliamente en el desarrollo del presente trabajo.

En el mercado existen varias posibilidad para la implementación de centrales telefónicas de voz, una de ellas es Asterisk, el cual es un software de código libre bajo licencia GPL (*General Public License*), esta licencia es la más ampliamente usada en el mundo del software y garantiza a los usuarios finales la libertad de usar, estudiar, compartir (copiar) y modificar el software.

Este software basado en Linux proporciona funcionalidad como cualquier central telefónica e incluye muchas características que anteriormente sólo estaban disponibles en costosos sistemas propietarios PBX, como buzón de voz, conferencias, Respuesta de voz interactiva (*IVR*), distribución automática de llamadas, entre otras.

En el presente trabajo se implementa una central telefónica basada en Asterisk con diferentes aplicaciones disponibles en las tiendas de aplicaciones de los diferentes sistemas operativos de mayor penetración a nivel mundial.

En el capítulo 1 del presente trabajo se describe una breve introducción sobre los componentes de un sistema de voz IP basado en una central telefónica asterisk empleando diferentes aplicaciones de voz IP para dispositivos móviles.

En el capítulo 2 se describen los fundamentos y principios básicos de la telefonía fija, los avances y desarrollo de la telefonía móvil, una breve introducción a la internet, estructura y funcionamiento del protocolo de la Internet, descripción de los protocolos TCP y UDP, descripción de un sistema de voz IP, así como el procesamiento de una señal de voz para su empleo en un sistema de voz IP, los protocolos empleados en voz IP, una breve descripción de las funciones de una central telefónica y una descripción de los componentes de una



central telefónica basada en asterisk, por último se mencionan algunas aplicaciones de voz IP para dispositivos móviles disponibles en el mercado.

En el capítulo 3 se describe los requerimientos que se proponen para la implementación de un sistema de voz IP con dispositivos móviles, mencionando el equipamiento básico del sistema, el servidor empleado para la instalación de la central telefónica, la instalación del sistema operativo Linux distro CentOS 6.3, así como la instalación y configuración de la central telefónica Asterisk versión 1.8.16.0 y por último la instalación de aplicaciones de voz IP para dispositivos móviles.

En el capítulo 4 se muestran el proceso de registro de usuarios, el inicio, establecimiento y finalización de una conversación de voz, además se muestran algunas herramientas para la depuración y corrección de errores tanto en el protocolo de inicio de sesión (*SIP*), como en el protocolo de transmisión en tiempo real (*RTP*) para finalizar este capítulo se muestran los resultados obtenidos durante la realización de una llamada telefónica mediante el empleo de una aplicación para análisis de protocolos.

En el capítulo 5 se muestran las conclusiones y trabajos futuros del presente trabajo.

Capítulo 2.

## **MARCO TEÓRICO.**

### **2.1 La telefonía.**

La invención del teléfono fue un evento que revolucionó el mundo de las comunicaciones, en sus inicios apenas fue empleado pero actualmente se ha convertido en un instrumento imprescindible.

La red telefónica básica o red telefónica pública conmutada (*PSTN Public Switched Telephone Network*) fue creada para transmitir la voz humana. Por la naturaleza de la información a transmitir así como por la tecnología disponible en la época en que fue creada, es de tipo analógico.

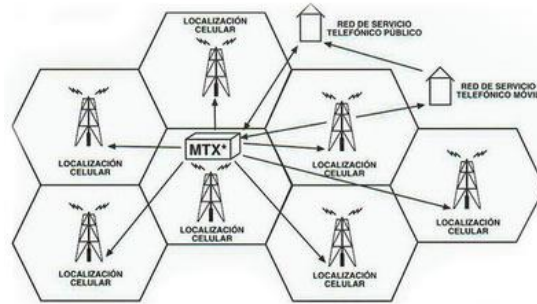
Aunque el teléfono lleva más de 130 años en operación, es hasta finales del siglo XX cuando las comunicaciones de voz sufren una evolución espectacular de la mano de dos fenómenos; la telefonía móvil e Internet los cuales se han conjugado para ofrecer todo un nuevo abanico de servicios muy atractivo e interesante.

Con la introducción de estas tecnologías, se facilita el acceso a alta velocidad a internet, permitiendo la oferta de múltiples servicios como voz, datos y televisión, por parte de los operadores telefónicos.

### **2.2 La telefonía móvil.**

La telefonía móvil es el empleo de un sistema celular que proporciona operación de teléfono estándar por radio de dos vías en lugares remotos.

El concepto básico detrás del sistema es que en vez de atender un espacio geométrico de amplias dimensiones, el área de servicios es dividida en áreas muy pequeñas denominadas *celdas* o *células*, como se muestra en la Figura 2.1.



**Figura 2.1 Diagrama de celdas en telefonía celular.**

Cada célula se conecta a líneas telefónicas o a enlaces de radio de microondas en un centro de control maestro, este controla todas las células y proporciona la interface entre cada célula y la oficina principal de teléfonos. Cuando un usuario pasa a través de una célula, se le atiende por el transceptor de la célula. La llamada del teléfono se coloca en ruta a través del control maestro y al sistema telefónico estándar. A medida que el usuario se mueve, el sistema conmuta automáticamente entre células. El receptor de cada estación de célula monitorea en forma continua el nivel de la señal de la unidad móvil. Cuando baja el nivel de la señal de un valor deseado, busca de manera automática una célula donde la señal de la unidad móvil sea más fuerte. La computadora del control maestro logra que la transmisión del usuario sea conmutada de la célula débil a la célula más fuerte. Esto se llama transferencia. Todo lo anterior sucede en un periodo muy corto y es por completo ignorado por el usuario. El resultado es la obtención de una transmisión y recepción óptimas.[1]

El paso inicial de las redes de telefonía móvil empezaron a ser operativas a principio de la década de los 80, cuya primera generación (1G) la constituyeron tecnologías analógicas como el AMPS (*Analog Mobile Phone System*) americano o el TACS (*Total Access Communication Systems*) europeo. Estas tecnologías estaban basadas en señalización analógica y eran orientadas únicamente a servicios de conmutación de circuitos, no soportando, por tanto, tráfico de datos.

Las primeras transmisiones de datos móviles fueron posibles gracias a la telefonía móvil de segunda generación (2G), pese que seguían orientadas al soporte del tráfico de voz.

El mayor exponente de la 2G es el estándar GSM (*Global System for Mobile communications*) el cual ha tenido gran aceptación a nivel mundial, de tal forma que un 75 % de dispositivos móviles cumplen esta norma.

Con la generación 2.5 , que incluye HSCSD (*High-Speed Circuit-Switched Data*), GPRS (*General Packet Radio Service*) y EDGE (*Enhanced Data rates for GSM Evolution*), se alcanzan velocidades de hasta 476 kbps proporcionando entre otros los servicios de mensajes texto (*Short Messaging Service o SMS*), mensajes multimedia (*Multimedia Messaging Service o MMS*) y aplicaciones de internet a través del *Wireless Application Protocol (WAP)*.

Actualmente se encuentran en operación los sistemas 3G, cuyo objetivo es definir un estándar mundial compatible con todos los sistemas anteriores, con voz de alta calidad y una velocidad de transmisión de datos de hasta 2 Mbps.

Por otro lado se han desarrollado dispositivos móviles con tecnología 4G los cuales emplean HSDPA (*High Speed Downlink Packet Access*) lo cual mejora la capacidad máxima de transferencia de información hasta alcanzar tasas de transferencia de hasta 14 Mbps.

De igual manera el campo de la electrónica ha permitido la evolución de los dispositivos móviles de comunicación, mejorando notablemente sus características.

Una característica con la que han sido dotados estos nuevos equipos es la posibilidad de conexión a redes de cómputo y por supuesto a Internet, esto nos permite establecer enlace de datos con cualquier parte del mundo que se encuentre conectado a esta red.

Esta interconexión se logra gracias a que muchos de estos nuevos dispositivos cuentan con tarjetas de red inalámbricas, lo cual permite tener acceso a una red de área local (*Local Area Network*) y de esta manera enlazar a los integrantes de esa red para envío y recepción de información entre ellos, si esta red cuenta con dispositivos de enrutamiento se puede interconectar con más redes como internet, esto permite que nuestro equipo de telefonía celular se pueda interconectar con cualquier otro dispositivo conectado a esta red y

empleando los protocolos adecuados para el establecimiento de la comunicación y el intercambio de información poder establecer comunicación.

Otra solución para la interconexión de equipos móviles de telefonía celular es el empleo de los canales de datos que proporciona el proveedor de telefonía celular quien se encarga de proporcionar una dirección IP al equipo y conectarse a internet.

### **2.3 Internet.**

Internet es la interconexión integrada de redes de ordenadores o redes interconectadas, esta red logra la interconexión de redes heterogéneas ya que está formada por una gran cantidad de redes más pequeñas y de las más diversas tecnologías.

La evolución de la telefonía ha venido acompañada e influenciada por el vertiginoso éxito de Internet. De hecho, la necesidad de tecnologías capaces de transportar y manejar tráfico de voz y datos en la misma infraestructura ha sido marcada precisamente por el éxito de la transmisión de datos.

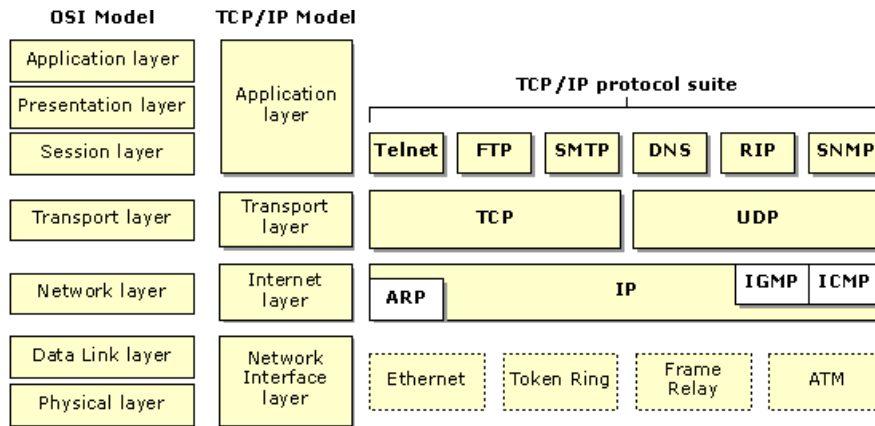
Con todo este desarrollo tecnológico los usuarios se encontraban con dos tipos de redes muy distintas; las redes de conmutación de circuitos para la transmisión de voz y las redes de conmutación de paquetes para la transmisión de datos, la idea principal era la unificación de ambas redes en una sola.

Al analizar la convergencia de ambas redes se optó por utilizar como tecnología de transporte por excelencia la conmutación de paquetes, y más concretamente el protocolo IP (*Internet Protocol*) en combinación con TCP (*Transport Control Protocol*). Las características de este protocolo, lo han convertido en la plataforma por excelencia de servicios, aplicaciones y tecnologías.

Por medio de un conjunto de componentes de hardware y software se crearon y continúan desarrollándose numerosos servicios, aplicaciones y usos de toda índole que son aprovechados para diferentes fines, los que conforman el infinito mundo de internet.

Gracias a la arquitectura TCP/IP para el usuario final, internet aparece como una única red.

Todo ello es posible gracias a una arquitectura de protocolos muy flexible dividida en cuatro niveles o capas, las cuales se muestran en la Figura 2.2



**Figura 2.2 Arquitectura TCP/IP.**

En este esquema la capa fundamental es la capa de internet, cuyo protocolo principal es el protocolo de internet (*IP*) sobre este nivel tenemos el nivel de transporte que ofrece dos tipos de servicios a las capas de nivel superior: fiable u orientado a conexión TCP (*Transport Control Protocol*) y no fiable o no orientado a conexión UDP (*User Datagram Protocol*) y sobre esta capa se encuentran las aplicaciones finales para empleo de los usuarios como Telnet (aplicación para ingresar a otro equipo de manera remota), FTP (aplicación para transferencia de archivos), entre otros más.

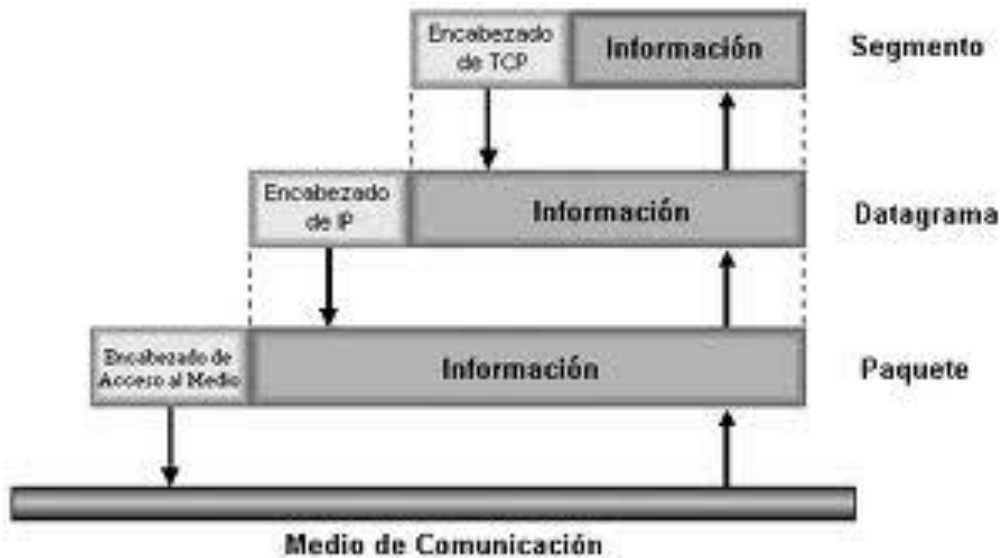
### 2.3.1 Protocolo IP.

Es el protocolo a nivel de red y ofrece un servicio sin garantía de servicio, del tipo conocido como de mejor esfuerzo (*best effort*).

En las redes IP la información se divide en fragmentos más pequeños denominados datagramas o paquetes, estos son enviados de manera independiente por la red, la información se envía sin un dialogo previo entre los extremos que garantice que la comunicación tendrá recursos suficientes para llevarse a cabo.

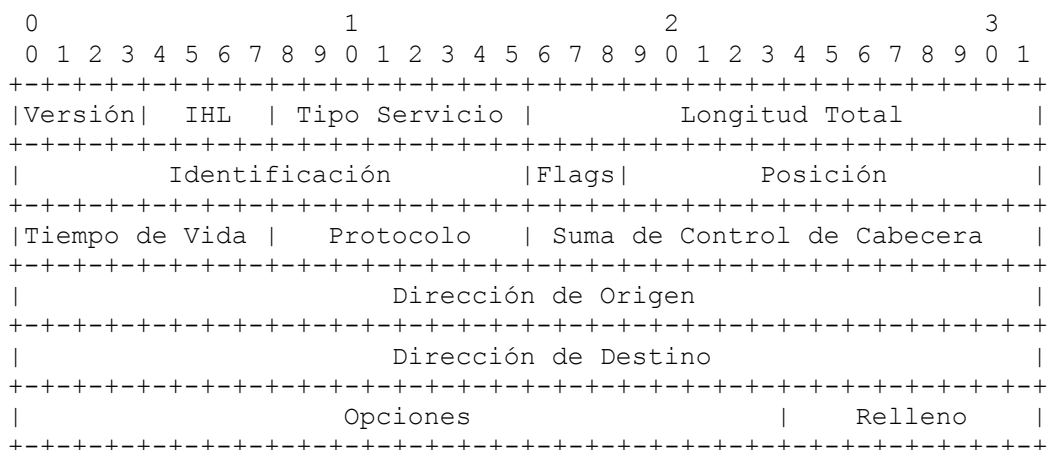
El tratamiento de esta manera tiene importantes implicaciones principalmente a la hora de transportar tráfico en tiempo real como voz, el hecho de que cada paquete curse un camino diferente para llegar a su destino no asegura que estos lleguen en el orden correcto, por lo que en el destino es necesario reordenarlos adecuadamente.

El paquete es la unidad mínima de información con la que se trabaja a nivel IP, este paquete encapsula el segmento de la capa de transporte que contiene, junto con la información de control pertinente, los datos de usuarios. Por otra parte, el paquete de IP se encapsulara, a su vez, en la trama de la subred concreta de que se trate, en la figura 2.3 se muestra el proceso de encapsulamiento de un paquete IP.



**Figura 2.3. Encapsulamiento de paquete IP.**

Un paquete IP se encuentra determinado por las características propias del protocolo, incluye información de direccionamiento así como para la reconstrucción de la información, la estructura general de un paquete IP se muestra en la Figura 2.4



**Figura 2.4 Estructura de paquete IP.**

### 2.3.2 Protocolo TCP.

Este protocolo proporciona un servicio fiable orientado a conexión, garantizando el flujo ordenado de bytes extremo a extremo con independencia del tipo y número de redes que atraviesen los paquetes, este protocolo lleva un control de errores y de flujo, además de un control preventivo de la congestión, todas estas características van a influir en el formato del paquete TCP, el cual se muestra en la figura 2.4

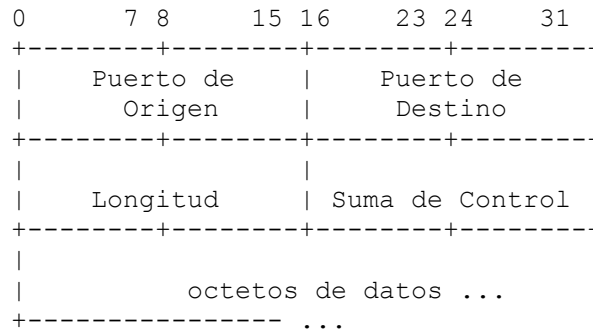


**Figura 2.5 Formato de paquete TCP.**



### 2.3.3 Protocolo UDP.

Ofrece un servicio no fiable, se emplea en aplicaciones sencillas que no requieran una alta fiabilidad o en entornos locales de alta fiabilidad con el fin de reducir los retardos asociados a TCP, debido a la simplicidad de este protocolo el formato del paquete UDP es muy sencillo, este se muestra en la figura 2.5



**Figura 2.6 Formato de paquete UDP.**

## 2.4 Introducción a la Voz IP (VoIP).

La voz IP o VoIP no es una tecnología nueva, su estudio y desarrollo se remonta a inicios de los década de los 90 cuando se generó el interés de transportar voz y video sobre redes IP como redes internas (*intranets*) e inclusive sobre internet. Esta tecnología está basada en la división de audio o video en pequeños fragmentos, transmitir dichos fragmentos a través de una red IP y re-ensamblar esos fragmentos en el destino final.

La voz IP significa básicamente que la voz viaja en Internet. Cuando se desarrolló VoIP por primera vez, funcionaba sólo y nada más que con Internet. En este momento, VoIP funciona en casi todos los tipos de redes, debido principalmente a la adopción del protocolo TCP/IP.

Aunque VoIP puede definirse de forma abreviada como una tecnología que aprovecha el protocolo TCP/IP para ofrecer conversaciones de voz, lo cierto es que es mucho más que esto, VoIP puede ser usada para reemplazar la telefonía tradicional en un entorno

empresarial, en un pequeño negocio o en casa, o simplemente para añadir ventajas a un sistema de telefonía tradicional.

En la arquitectura de VoIP se pueden distinguir los siguientes dispositivos:

- **Teléfono IP.** Es un teléfono similar a un teléfono tradicional con la diferencia que está adaptado para ser utilizado en entornos IP.
- **Softphone.** Es una aplicación que es instalada en un equipo de cómputo o en un dispositivo móvil para realizar comunicaciones de voz IP.
- **Adaptador Telefónico Analógico (ATA).** Es un adaptador que permite conectar un teléfono convencional a una red IP.
- **SIP (Session Initiation Protocol).** Es un protocolo usado por los proveedores de VoIP encargado de, entre otras funciones, iniciar y finalizar las llamadas VoIP.
- **B2BUA (Back to Back User Agent).** Es una entidad intermediaria encargada de procesar las comunicaciones VoIP y retransmitirlas a su destino.

En la figura 2.6 se muestra un diagrama de la arquitectura de VoIP

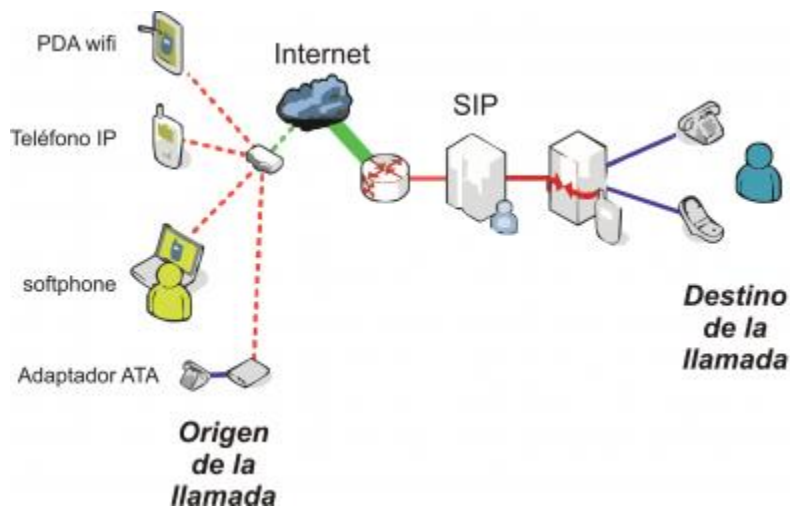


Figura 2.7 Arquitectura de Voz IP.

Partiendo que la voz es una señal analógica y un paquete IP es enteramente digital se deben hacer ciertos procesos para convertir la voz humana, la cual es una señal analógica, en un paquete IP, estos procedimientos son la base que siguen todos los protocolos.

La señal de voz proviene de una fuente acústica y esta a su vez es convertida en una señal eléctrica a través de un transductor.

En la práctica para la transmisión de voz digitalizada se emplean sistemas de ocho bits por muestra, lo que equivale a trabajar con 256 niveles, para telefonía normalmente se emplea la modulación por pulsos codificados o PCM (*Pulse Code Modulation*), donde cada muestra se representa por 8 bits, resultando un flujo de 64 kbps ( $8,000 \times 8$ ).

Existen tres clasificaciones de codificadores:

- De forma de onda.
- Vocoders.
- Híbridos.

Los de forma de onda engloban a los codificadores que intentan reproducir la forma de onda de la señal de entrada sin tener en cuenta la naturaleza de la misma. En función del dominio en que opere, se dividen en Codificadores en el dominio del tiempo y Codificadores en el dominio de la frecuencia.

Este tipo de codificadores proporcionan una alta calidad de voz a tasas de transferencia media. Sin embargo, no son útiles cuando se quiere codificar a niveles bajos (cuando no se dispone del suficiente ancho de banda), un ejemplo de esto es que para el códec G711 se usa PCM, lo cual lo hace de buena calidad pero no apropiado para el uso a través de internet.

Los vocoders si tienen en cuenta la naturaleza de la señal a codificar, en este caso la voz, y aprovechan las características de la misma para ganar en eficiencia. Permiten trabajar con muy bajas tasas de transferencia, pero la señal de voz que producen suena demasiado sintética.

Los codificadores híbridos combinan técnicas de los vocoders y de los codificadores de forma de onda permitiendo una alta calidad de voz a bajas tasas de transferencia.

En estos codificadores, las muestras de la señal de entrada se dividen en bloques de muestras (vectores) que son procesados como si fueran uno solo. Llevan a cabo una representación paramétrica de la señal de voz para tratar que la señal sintética se parezca lo más posible a la original.

También se les conoce como codificadores de análisis por síntesis. En el emisor se lleva a cabo un análisis que obtiene los parámetros de la señal para luego sintetizarla y conseguir el mayor parecido a la original.

En la tabla 2.1 se muestran los codecs (*codificador-decodificador*) más comunes usados para telefonía.

Nombre	Estandarizado	Descripción	Bit rate (kb/s)	Sampling rate (kHz)	Frame size (ms)	Observaciones	MOS (Mean Opinion Score)
<a href="#">G.711 *</a>	ITU-T	Pulse code modulation (PCM)	64	8	Muestreada	Tiene dos versiones u-law (US, Japan) y a-law (Europa) para muestrear la señal	4.1
<a href="#">G.711.1 * -</a>	ITU-T	Pulse code modulation (PCM)	80-96Kbps	8	Muestreada	Mejora del codec G.711 para abarcar la banda de 50 Hz a 7 Khz. <a href="#">Mas info</a>	
<b>G.721</b>	ITU-T	Adaptive differential pulse code modulation (ADPCM)	32	8	Muestreada	Obsoleta. S e ha transformado en la G.726.	
<b>G.722</b>	ITU-T	7 kHz audio-coding within 64 kbit/s	64	16	Muestreada	Divide los 16 Khz en dos bandas cada una usando ADPCM	
<b>G.722.1</b>	ITU-T	Codificación a 24 y 32 kbit/s para sistemas sin manos con baja perdida de paquetes	24/32	16	20		
<b>G.722.2 AMR-WB</b>	ITU-T	Adaptive Multi-Rate Wideband Codec (AMR-WB)	23.85/ 23.05/ 19.85/ 18.25/ 15.85/ 14.25/ 12.65/ 8.85/ 6.6	16	20	Se usa principalmente para compresion de voz en tecnología movil de tercera generación. <a href="#">Mas info</a>	
<b>G.723</b>	ITU-T	Extensión de la norma G.721 a 24 y 40 kbit/s para aplicaciones en circuitos digitales.	24/40	8	Muestreada	Obsoleta por G.726. Es totalmente diferente de G.723.1.	
<b>G.723.1</b>	ITU-T	Dual rate speech coder for multimedia communications transmitting at 5.3 and	5.6/6.3	8	30	Parte de H.324 video conferencing. Codifica la señal usando linear predictive analysis-by-synthesis coding. Para el codificador de high rate utiliza Multipulse Maximum	3.8-3.9

		6.3 kbit/s				Likelihood Quantization (MP-MLQ) y para el de low-rate usa Algebraic-Code-Excited Linear-Prediction (ACELP).	
<b>G.726</b>	ITU-T	40, 32, 24, 16 kbit/s adaptive differential pulse code modulation (ADPCM)	16/24/32/40	8	Muestreada	ADPCM; reemplaza a G.721 y G.723.	3.85
<b>G.727</b>	ITU-T	5-, 4-, 3- and 2-bit/sample embedded adaptive differential pulse code modulation (ADPCM)	var.		Muestreada	ADPCM. Relacionada con G.726.	
<b>G.728</b>	ITU-T	Coding of speech at 16 kbit/s using low-delay code excited linear prediction	16	8	2.5	CELP.	3.61
<a href="#">G.729 **</a>	ITU-T	Coding of speech at 8 kbit/s using conjugate-structure algebraic-code-excited linear-prediction (CS-ACELP)	8	8	10	Bajo retardo (15 ms)	3.92
<a href="#">G.729.1</a>	ITU-T	Coding of speech at 8 kbit/s using conjugate-structure algebraic-code-excited linear-prediction (CS-ACELP)	8/12/14/16/18/20/22/24/26/28/30/32	8	10	Ancho de banda desde 50Hz a 7 KHz	
<b>GSM 06.10</b>	ETSI	RegularPulse Excitation LongTerm Predictor (RPE-LTP)	13	8	22.5	Usado por la tecnología celular GSM	
<b>LPC10</b>	Gobierno de USA	Linear-predictive codec	2.4	8	22.5	10 coeficientes.La voz suena un poco "robotica"	
<b>Speex</b>			8, 16, 32	2.15-24.6 (NB) 4-44.2 (WB)	30 ( NB ) 34 ( WB )		
<b>iLBC</b>			8	13.3	30		
<b>DoD CELP</b>	American Department of Defense (DoD) Gobierno de USA		4.8		30		
<b>EVRC</b>	3GPP2	Enhanced Variable Rate CODEC	9.6/4.8/1.2	8	20	Se usa en redes CDMA	
<b>DVI</b>	Interactive Multimedia Association (IMA)	DVI4 uses an adaptive delta pulse code modulation (ADPCM)	32	Variable	Muestreada		
<b>L16</b>		Uncompressed audio data samples	128	Variable	Muestreada		
<b>SILK</b>	Skype	Uncompressed audio data samples	De 6 a 40 kbit/s	Variable	20	El codec Harmony está basado en SILK	

**Tabla 2.1 Codecs más usados para telefonía.**

### **2.4.2 Telefonía conmutada por circuitos.**

Antes que las redes digitales despegaran, todos teníamos que utilizar el único sistema telefónico antiguo y sencillo, este sistema funciona en una red llamada Red Pública de Telefónica Conmutada (*PSTN Public Switched Telephone Network*).

Estos sistemas telefónicos utilizan el antiguo método conocido como conmutado por circuitos.

### **2.4.3 Telefonía conmutada por paquetes.**

A diferencia de los conmutados por circuitos, que siempre requieren el uso de la Red Pública Telefónica Conmutada (PSTN), la tecnología VoIP ha permitido que la telefonía y otras características y servicios nuevos funcionen en redes dedicadas e inalámbricas incluyendo hasta su red casera. Estos tipos de red más nuevos utilizan protocolos de paquetes conmutados.

El sistema conmutado por paquetes VoIP convierte señales de voz en paquetes de datos, junto a las señales de voz, los paquetes VoIP incluyen tanto las direcciones de red del emisor como las del receptor y pueden recorrer cualquier red compatible con VoIP. Luego, pueden elegir caminos alternos ya que la dirección de destino está incluida en el paquete, la ruta de los paquetes no depende de ninguna ruta de red en particular.

En una red conmutada por circuitos, la dirección de destino no se incluye en la señal; las direcciones de ruteo están determinadas físicamente por la línea real, por lo que la ruta debe seguir una línea de red, si la línea está caída, la llamada no puede continuar.

En una red conmutada por paquetes, si una de las líneas de red está caída, el paquete se puede conectar mientras está entre ubicaciones a otra ruta funcional para mantener la llamada activa. Al utilizar VoIP, las señales de voz se pueden separar en paquetes tal como los paquetes de datos informáticos. Esto hace que las compañías consideren utilizar la misma infraestructura de red para mantener aplicaciones de voz y de datos.

La VoIP proporciona otros servicios que los sistemas de telefonía más antiguos no pueden proporcionar. Los protocolos VoIP, o simplemente *IP*, son interoperables, lo que significa que trabajarán bien con todo tipo de redes. Los protocolos IP también son altamente portátiles. Esto quiere decir que trabajarán con cualquier dispositivo de usuario final habilitado para IP tal como un teléfono IP, una computadora, un teléfono celular, etc.

La Telefonía IP permite la comunicación de voz mediante las redes del Protocolo de Internet (IP). Une muchas de las localidades de una organización -incluyendo a los trabajadores móviles- dentro de una sola red convergente.

Para respaldar la Telefonía IP, en una LAN un servidor se dedica por lo general a cargar el software que se utiliza para administrar todas las llamadas. El servidor encargado de la administración almacena la base de datos que contiene todas las direcciones de control de acceso al medio MAC (*Media Access Control*) de las extensiones de Telefonía IP que se asignarán a los usuarios finales. Dependiendo del tamaño de la red, uno o más dispositivos conocidos como conmutadores o switches se instalan a su alrededor para formar la infraestructura nuclear de la Telefonía IP.

Cuando desea llamar a un usuario en su misma red, marca el número telefónico correspondiente. Las señales se empaquetan y se envían al servidor encargado de la administración en donde el paquete recoge la dirección MAC de la persona a la que está llamando. El paquete se reenvía al conmutador, al puerto en ese conmutador y finalmente al teléfono IP conectado al puerto que corresponde a la dirección MAC de su compañero. Cuando contesta la llamada, una conexión virtual se establece entre ambos usuarios durante la llamada.

El proceso varía un poco cuando se llama a un usuario que se encuentra en algún lugar fuera de su red local. Este está conectado a una red diferente. La llamada comenzaría del mismo modo, pero en lugar de que el administrador lo envíe a un interruptor localizado en su red, la llamada necesita ir al dispositivo de frontera utilizado para conectar su red a la red donde se encuentra el otro usuario. Aquí es donde se convierte a VoIP.

A diferencia de la dirección MAC en el lado LAN, el tráfico VoIP en la WAN utiliza un esquema de direcciones IP. La dirección IP en este momento utiliza un formato de 4 octetos y representa la dirección en números decimales. Por ejemplo, 192.168.2.4 es una dirección IP válida. Cuando los paquetes llegan al destino LAN, el dispositivo de frontera divide los paquetes VoIP y los reenvía de forma interna al servidor que administra los servicios de Telefonía IP en la LAN. A partir de este punto, el resto del proceso es similar a los servicios de Telefonía IP. El teléfono suena; la persona contesta y un circuito virtual se establece entre ambos.

#### **2.4.4 Protocolos.**

Un protocolo es un conjunto de reglas usadas por computadoras para comunicarse unas con otras a través de una red por medio de intercambio de mensajes. Puede ser definido como las reglas o el estándar que define la sintaxis, semántica y sincronización de la comunicación. Los protocolos pueden ser implementados por hardware, software, o una combinación de ambos. A su más bajo nivel, define el comportamiento de una conexión de hardware.

Para VoIP primeramente hace su aparición el protocolo de señalización H323 el cual fue estandarizado por la ITU-T, siendo aún utilizado en muchos sistemas de VoIP, luego aparece el protocolo SIP estandarizado por la IETF el cual es ampliamente usado en los sistemas actuales.

##### **2.4.4.1 Inicio de Sesión.**

Existen varios tipos de protocolos para inicio y control de sesión, aunque los de mayor uso son el H323, SIP Y IAX.

En este trabajo nos enfocaremos el protocolo SIP por ser el de mayor difusión y empleo a nivel mundial, así como en el que está basada la solución propuesta.



El protocolo SIP es la pieza fundamental en la arquitectura multimedia de la IEF, la versión más actual es la SIP versión 2 la cual apareció en el 2002, se encuentra publicado en el RFC 3261[3]

Este protocolo es a nivel aplicación y se ocupa del establecimiento, modificación y finalización de una sesión multimedia, las cuales se llevan a cabo de manera interactiva, está basado en un modelo de conexión cliente servidor y es altamente flexible.

El protocolo de inicio de sesión utiliza el protocolo de transporte UDP con el puerto 5060 por defecto pero también se puede utilizar TCP, para establecimiento de sesiones utiliza el protocolo SDP; para realizar una sesión VoIP se usan 2 canales uno para señalización y otro que es el canal de flujo de datos de carga útil o stream, en la figura 2.8 se ejemplifica el esquema de este protocolo.



**Figura 2-8 Esquema de protocolo SIP.**

El protocolo SIP es basado en texto, muy similar al Protocolo de Transferencia de Hipertexto (*HTTP Hypertext Transfer Protocol*) por la forma en que funciona (basado en texto) y similar al Protocolo de Transferencia Simple de correo electrónico Simple (*SMTP Simple Mail Transfer Protocol*) en la forma en que se especifican las direcciones SIP este genera mensaje de petición y respuesta

Las direcciones SIP identifican a un usuario de un determinado dominio. A estas direcciones SIP habitualmente se les llama **URI (Uniform Resource Identifier)**. Una URI se puede especificar de las siguientes maneras [3]:

[sip:usuario@dominio\[:port\]](#)

[sip:usuario@direcciónIP\[:port\]](#)

El dominio representa el nombre del proxy SIP que conoce la dirección IP del terminal identificado por el usuario de dicho dominio. El puerto por defecto para SIP es 5060, aunque es posible especificar otros adicionales si es necesario.

Por ejemplo si tenemos registrado el usuario 510 perteneciente al dominio ipn.mx, el usuario SIP, estaría conformado de la siguiente manera:

[510@ipn.mx](mailto:510@ipn.mx)

Por otro lado el usuario 510 perteneciente al dominio con dirección IP 148.205.25.170 quedaría de la siguiente manera:

510@148.205.25.170

Una de las grandes ventajas de este protocolo es que la localización de los usuarios está basada en los sistemas de nombre de dominio de nombre (*DNS Domain Name System*), además de ser compatible con el protocolo H323.

Los elementos necesarios para establecer una comunicación SIP son los siguientes:

- **Agentes Usuario** (*User Agents UA*) o clientes SIP, estas entidades manejan la señalización SIP, se encuentra divididos en dos agentes usuarios:
  - **User agent client (UAC)** es el elemento que realiza peticiones SIP y acepta respuestas del elemento que lo complementa (*UAS*), en nuestro caso es el dispositivo móvil.
  - **User agent server (UAS)** es el elemento que recibe las peticiones SIP realizadas por el UAC y envía las respuestas convenientes, este elemento de igual forma puede ser nuestro dispositivo móvil, debido a que acepta peticiones de inicio de comunicación enviadas por otro móvil. Un servidor SIP también es un UAS.

- **Intermediarios o servidores**, estos elementos son necesarios para que la comunicación entre dos UA pueda llevarse a cabo, estos elementos pueden ser los siguientes:
  - **Servidores Proxy (Proxy server)**. Este forma la parte más importante del servicio y que se encarga de la carga de las llamadas y de re direccionar los datos según sea el caso a cada uno de los otros servidores.
  - **Servidores de Localización (Local Server)** Es usado por el servidor Proxy o por el servidor de reenvío para obtener información de la posible localización de un cliente SIP al cual se lo está llamando, este puede estar de manera externa a la red SIP.
  - **Servidores de Registro (Register server)** A este servidor se envían peticiones de registro por parte de los clientes para actualizar ubicación y registro de cada uno.
  - **Servidor de Redirección (Redirection Server)**. Permite el establecimiento de llamadas entre clientes sin usar el proxy, esto disminuye la carga a los servidores Proxy y es usado cuando se tiene llamadas concurrentes.
  - **Modo persistente (Stateful Proxy server)** El servidor asigna a cada petición entrante una petición saliente, creando para ello un nuevo proceso de usuario que atenderá a cada llamada entrante.
  - **Servidores Proxy de Estado Volátil (Stateless Proxy Server)**. Procesa cada petición entrante en modo secuencial y elimina cualquier resto de la misma una vez que la petición saliente se ha generado. Consume menos recursos.

El protocolo SIP puede proveer los siguientes servicios

- **Servicio de localización de usuarios:** determinación de las direcciones de localización de los usuarios finales que participaran en las sesiones multimedia.
- **Servicio de establecimiento de llamada:** repique, negociación de parámetros entre los participantes de una sesión.
- **Servicio de Disponibilidad de Usuario:** posibilidad de determinar si un agente SIP desea o no participar en una nueva sesión.
- **Servicio de características de usuario:** posibilidad de determinar los flujos multimedia, y las características de los mismos, que podrán utilizarse en una sesión.
- **Servicio de Manejo de Llamadas:** transferencia de llamadas y terminación de sesiones.

Los mensajes que se intercambian entre los distintos elementos durante una comunicación, para realizar las funciones de este protocolo se muestran en la tabla 2.2

Petición SIP	Descripción
<b>INVITE</b>	Es la petición SIP que se envía a un usuario cuando queremos establecer comunicación con él.
<b>ACK</b>	Esta petición es enviada por el usuario origen que envió la petición INVITE para hacer saber al usuario destino que su respuesta ha sido recibida. Es el momento en que ambos pueden empezar a enviar tráfico media.
<b>BYE</b>	Para finalizar la comunicación entre los dos usuarios establecida anteriormente con INVITE.
<b>CANCEL</b>	Se utiliza para cancelar una petición, por ejemplo INVITE, que se encuentra en progreso. Por ejemplo si el teléfono destino está sonando pero aún no ha sido descolgado y el teléfono origen cuelga, se envía un CANCEL a diferencia de un BYE que se enviaría si el teléfono destino hubiera sido descolgado previamente y por tanto la comunicación establecida unos instantes.
<b>OPTIONS</b>	Un UA puede enviar peticiones OPTIONS a un UAS para solicitar cierta información sobre este

<b>REGISTER</b>	Un UAC envía peticiones REGISTER a un servidor de registro localización para informar de la posición actual en la que se encuentra en un momento determinado. Esto hace posible que el UAC pueda ser localizado haciendo uso de su misma dirección user@dominio sin importar donde el UAC se encuentre físicamente.
-----------------	---

**Tabla 2.2 Peticiones del protocolo SIP.**

Cada petición SIP lleva asociada la respuesta correspondiente enumerada mediante un código identificador, las cuales se presentan en la tabla 2.3

Tipo de Respuesta	Identificador	Significado
Informan del estado provisional de la comunicación.	100	Trying-Intentando
	180	Ringing- Sonando
	181	Call Being Forwarded-Llamada está siendo transferida
	182	All Queued-Encolada
	183	Sesion Progress-Llamada en progreso
Informan del éxito de la comunicación.	200	OK-OK
	202	Accepted- Aceptada
Informan del reenvío necesario de la petición SIP	300	Multiple Choices – Multiples opciones
	301	Moved Permanently –Movido permanentemente
	302	Moved Temporarily – Movido temporalmente.
	305	Use Proxy – Usar proxy
	380	Alternative Service – Servicio Alternativo
Informan de errores del cliente	400	Petition Bad request – Mala petición.
	401	Unauthorized – No autorizado
	402	Payment Required – Se requiere pago
	403	Forbidden - Prohibido
	404	Not Found – No encontrado
	405	Method Not Allowed –Método no permitido.
	406	Not Acceptable – No es aceptable.
	407	Proxy Authentication Required – Se requiere autenticación
	408	Request Timeout – Tiempo agotado para la petición
	410	Gone – Se ha marcado
	413	Request Entity Too Large – Petición demasiado grande
414	Request URI Too Long – URI demasiado larga	

	415	Unsupported Media Type – Tipo de media no soportado
	416	Unsupported URI Scheme – Esquema URI no soportado
	420	Bad extensión – Extensión incorrecta
	421	Extension Required – Se requiere extensión
	423	Interval Too Brief – Intervalo demasiado corto
	480	Temporarily Unavailable – No disponible temporalmente
	481	Call/Transaction Does Not Exist – No existe la llamada/Transacción
	482	Loop Detected – Circulo vicioso detectado
	483	Too Many Hops – Demasiado Hops
	484	Address Incomplete – Dirección Incompleta
	485	Ambiguous - Ambiguo
	486	Busy Here - Ocupado
	487	Rqueste terminated – Petición terminada
	488	Not Acceptable Here – No es aceptable aquí
	491	Request Pending – Petición pendiente
	493	Undecipherable – Indescifrable
Informan de errores del servidor	500	Server Internal Error – Error Interno del servidor
	501	Not Implemented – No implementado
	502	Bad Gateway – Gateway incorrecto
	503	Service Unavailable – Servicio no disponible
	504	Service Time-out – Tiempo agotado en el servidor
	505	Version Not Supported – Version no soportada
	513	Message Too Large – Mensaje demasiado largo
Informan de errores generales	600	Busy Everywhere – Ocupado en todos sitios
	603	Declined - Rechazado
	604	Does Not Exist Anywhere – No existe en ningún sitio.
	606	Not Acceptable – No aceptable-

**Tabla 2.3 Respuestas del protocolo SIP.**

Conocidos los mensajes y posibles respuestas SIP, en la figura 2.11 se ejemplifica el proceso de establecimiento de una llamada.

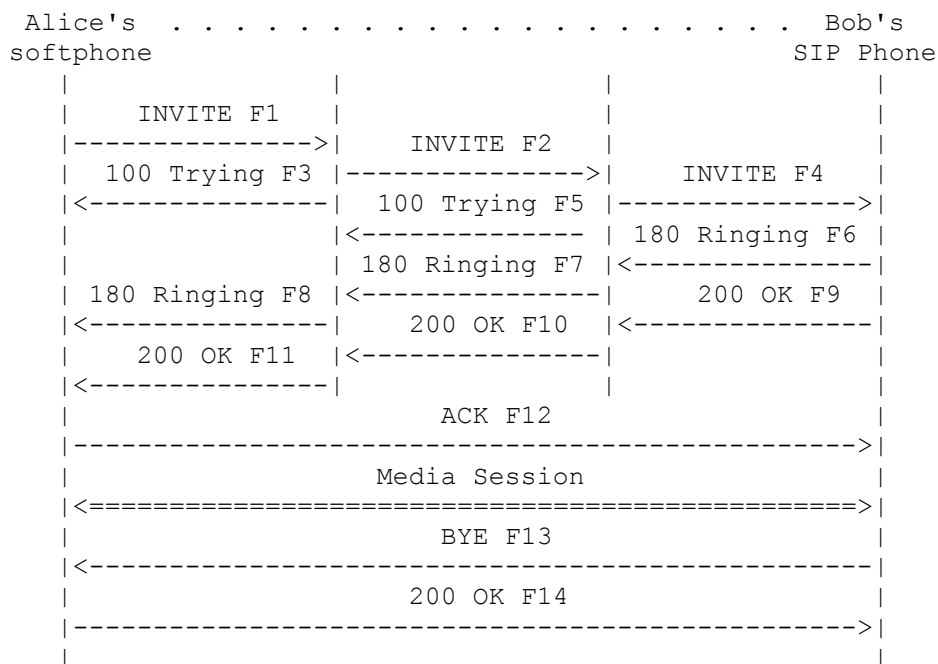


Figura 2.9 Proceso de establecimiento de llamada entre dos usuarios.

#### 2.4.4.2 Audio.

Se ha mostrado el protocolo SIP para establecer una comunicación entre dos extremos, sin embargo como ya se indicó anteriormente esto no es suficiente para establecer una comunicación multimedia. Para establecer un flujo de comunicación multimedia es necesario un protocolo que intercambie la información entre los extremos de dicha comunicación, es decir, que transporte la información entre un origen y su destino, además de proveer de las técnicas necesarias para enviar los problemas que se pueden presentar durante el intercambio, tales como: jitter, retardo, etc.

Los protocolos más utilizados para el transporte de audio y video en tiempo real son:

- **Protocolo en Tiempo Real (RTP Real Time Protocol).**
- **Protocolo de Control en Tiempo Real (RTCP Real Time Control Protocol).**

RTP es el encargado de transportar tanto audio como vídeo en tiempo real. Utiliza UDP como protocolo de transporte, ya que el uso de TCP daría lugar a un retardo elevado durante la comunicación a causa de las retransmisiones. Este protocolo hace uso de un número de secuencia, marcas de tiempo, envío de paquetes sin retransmisión, identificación del origen, identificación del contenido, sincronización, etc. para continuar con la reproducción del flujo de paquetes. Por lo tanto no puede garantizar que la entrega de tráfico se haga en tiempo real, aunque sí garantiza al menos que lo hará de forma sincronizada.

El protocolo RTCP se encarga de monitorear el flujo de los paquetes RTP. Obtiene estadísticas relacionadas con la calidad de servicio. El inconveniente es que, aunque realice un monitoreo de la calidad de servicio de RTP, no se poseen mecanismos como reservar ancho de banda o control de la congestión para intentar resolver una situación en la que la calidad de la transmisión no es suficiente. Es por ello por lo que la utilización de RTCP es opcional, aunque también recomendable.

El protocolo SIP se integra perfectamente con protocolos de audio como RTP, RTSP (*Real Time Streaming Protocol*) RFC 2326[4] gracias al protocolo SDP (*Session Description Protocol*) RFC 3266 [5], el cual, permite formar una completa arquitectura multimedia, este se utiliza durante la negociación que lleva a cabo el protocolo de inicio de sesión entre los dos agentes. Su función es la de detallar cómo se va a realizar el intercambio de comunicación posterior mediante protocolos como, por ejemplo, RTP. Para ello indica toda la información relacionada con el tráfico multimedia tal como la dirección IP y el puerto donde espera cada agente recibir el audio, el códec a utilizar entre ambos, etc.

## 2.5 Central Telefónica Asterisk

Una central telefónica es un dispositivo que permite a los usuarios u abonados compartir un determinado número de líneas externas para hacer llamadas telefónicas entrantes o



salientes, así como establecer comunicaciones internas entre todos los dispositivos que dependen de la PBX. Una de las ventajas que ofrece una PBX es una solución mucho menos cara que proporcionar a cada usuario de la empresa o institución una línea telefónica externa. Así mismo, a una PBX se le pueden conectar teléfonos, máquinas de fax, módems y otros dispositivos de comunicación.

En el mercado existen soluciones basadas en hardware, las cuales implementan los dispositivos con protocolos nativos propietarios, por lo cual no existe la interoperabilidad entre dispositivos con diferentes protocolos, algunos de estos ejemplos son los conmutadores HARRIS, HYBEX, SIEMENS, CISCO, PANASONIC, NORTEL, etc. Además que estas soluciones son un tanto costosas.

Una de las soluciones actuales es la implementación de centrales telefónicas basadas en software de código libre, por ejemplo FreeSWITCH y Asterisk.

Asterisk es una central telefónica de software libre que se distribuye bajo licencia GNU. Este producto, soportado comercialmente por Digium, se ha convertido en pocos años en una de las soluciones IP más extendidas en diversos ámbitos, como el empresarial o el educativo.

Asterisk fue diseñado de manera modular, de manera que cada usuario pueda seleccionar qué partes de Asterisk o módulos desea utilizar. Esto hace de Asterisk una aplicación realmente escalable y extensible

- Escalable. Ya que es posible activar o desactivar los módulos no utilizados para instalar Asterisk en dispositivos embebidos de pocos recursos.
- Extensible. Para programar un nuevo módulo de Asterisk no es necesario conocer todo el código de Asterisk.

En la fig. 2.12 se muestra el diseño modular de Asterisk .



**Figura 2.10** Diseño modular de Asterisk.

Los módulos de Asterisk se dividen en 7 categorías:

- **Core.** Se trata del núcleo de Asterisk, que incluye las funciones más básicas y posibilita la carga de módulos.
- **Recursos.** Aportan funcionalidades adicionales al core, como la posibilidad de leer ficheros de configuración (res\_config), música en espera (res\_musiconhold), etc.
- **Canales.** Permiten a Asterisk manejar dispositivos de una determinada tecnología. Por ejemplo, para manejar dispositivos SIP se utiliza el módulo chan\_sip, para IAX2 chan\_iax y para canales analógicos/digitales chan\_zap.
- **Aplicaciones y funciones.** Estos módulos conforman la "caja de herramientas" de Asterisk, ya que son los módulos que aportan las distintas herramientas para configurar nuestro sistema Asterisk.
- **CDR.** Estos módulos controlan la escritura del registro telefónico generado por Asterisk a diferentes formatos, por ejemplo a un archivo o a una base de datos.
- **Codecs.** Para que Asterisk pueda codificar y decodificar la información de audio/vídeo que tiene que enviar y recibir dispone de distintos codecs.

- **Formatos.** Estos módulos posibilitan a Asterisk entender y manejar archivos en distintos formatos, como mp3, alaw, ulaw, etc.

La configuración de un servidor Asterisk se puede hacer desde dos puntos:

- Desde el propio CLI (*Interface de líneas de comando*)
- Desde los ficheros de configuración (*.conf*) ubicados en `/etc/asterisk`

Para la configuración de Asterisk se han implementado objetos mediante una Interfaz Gráfica de Usuario (*GUI Grafic User Interface*) con lo que se minimiza en gran medida la configuración ya que se ocupan menús y estos a su vez escriben dentro de los archivos de configuración

Un ejemplo de estas aplicaciones GUI son

- AsteriskNOW
- Trixbox (FreePBX)

Estas aplicaciones vienen con una versión de Linux muy ligera para que los recursos sean mayormente ocupados por la aplicación, ya que el uso de codecs de alta compresión hacen que la carga de procesamiento sea alta.

Asterisk es una aplicación basada en Linux, pero no en una distribución específica, podemos encontrar información respecto a este tema principalmente en dos distribuciones, Debian y CentOS.

## 2.6 Aplicaciones de VoIP para dispositivos móviles.

Actualmente existen una infinidad de aplicaciones de VoIP para dispositivos móviles, en función del sistema operativo bajo el que se encuentran operado el dispositivo. Los dispositivos móviles funcionan en tres sistemas operativos principalmente, Android, iOS y el Blackberry OS.

Estos dos últimos sistemas operativos son nativos de marcas propietarias Mac y Blackberry.

El desarrollo de aplicaciones para estos sistemas operativos actualmente ha tenido un auge sin precedente, lo cual ha permitido el desarrollo de aplicaciones en todos los ámbitos, desde entretenimiento, aplicaciones de oficina, estado del tiempo, etc.

Las aplicaciones de voz IP no han sido la excepción, aunque en este tema el desarrollo de aplicaciones para el sistema operativo Blackberry OS se ha mantenido muy por debajo de la expectativa, todo ello debido a las restricciones impuestas por el mismo sistema operativo generando que los desarrolladores de aplicaciones no incursionen en la investigación y desarrollo de aplicaciones de este tipo. El desarrollo se ha realizado principalmente en países del continente europeo como Inglaterra, Italia y Francia.

En este trabajo realizaremos el análisis de aplicaciones desarrolladas para los sistemas operativos mencionados con anterioridad, ya que hasta el momento no existe documentación que muestre el resultado de desempeño de algunas de estas aplicaciones.

En la figura 2.13 se muestran la iconografía de algunas aplicaciones de voz IP disponibles en el mercado para los diferentes sistemas operativos en estudio, compatibles con los protocolos SIP y RTP:



**Figura 2.11 Aplicaciones de voz IP para dispositivos móviles.**

A continuación realizaremos un breve análisis de estas aplicaciones y determinaremos cuál de ellas es la que emplearemos para la implementación del sistema.

### **2.6.1 VMobile**

Esta aplicación se encuentra disponible en la tienda de descargas Blackberry world, actualmente al descargarla se instala una versión de prueba con una duración de 30 días, al concluir este periodo de prueba es necesario adquirir mediante una nueva licencia.

Esta es una aplicación muy intuitiva, que una vez conociendo el funcionamiento de una central telefónica de voz IP es fácilmente configurable, únicamente cuenta con el codec nativo de Blackberry (AMR) sus principales desventajas son el uso de un codec exclusivo y el costo de la aplicación.

### **2.6.2 Join Express**

Esta aplicación es muy intuitiva, con parámetros adicionales para configurar, cuenta con diferentes codecs seleccionables PCM (ulaw y alaw) y AMR , esta es una aplicación completamente gratuita.

### **2.6.3 Linphone**

Esta es una aplicación de código libre disponible para diversos sistemas operativos como Android, iOS y Blackberry, la versión disponible en el Blackberry world es completamente gratuita, una de sus principales desventajas es que únicamente se encuentra implementado el codec AMR, esta es una excelente aplicación si deseamos implementar el sistema con diversos sistemas operativos.

Por lo anteriormente expuesto se seleccionó la aplicación cliente de voz IP Join Express para Blackberry, debido a que es gratuita y cuenta con una mayor cantidad de codecs disponibles.

La instalación de la aplicación es muy simple e intuitiva, solo debemos ingresar a la aplicación de descarga de aplicaciones de Blackberry, Blackberry world buscamos Join VoIP, seleccionamos el icono respectivo e iniciamos la descarga e instalación de esta aplicación.

## Capítulo 3

# MODELO PROPUESTO.

### 3.1 Descripción del modelo.

Para la implementación de un sistema de voz IP para dispositivos móviles, es importante considerar que a la fecha las empresas desarrolladoras de dispositivos móviles no cuentan con aplicaciones propietarias de voz IP.

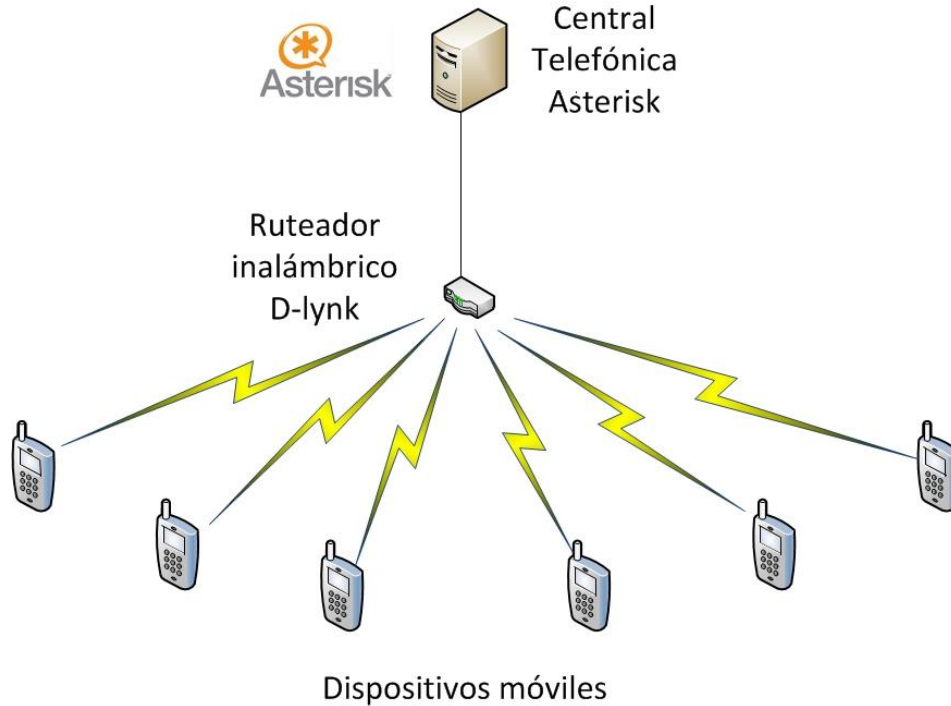
Por lo que anterior, diversas empresas desarrolladoras de tecnología han incursionado en el desarrollo de aplicaciones de voz IP para dispositivos móviles.

Durante la etapa de investigación motivo de este trabajo de tesis se realizaron pruebas de funcionamiento a diversas aplicaciones para conocer sus características y de esta manera seleccionar las que proporcionen las mayores ventajas para la implementación del sistema en estudio.

Por otro lado la central telefónica que se consideró para la implementación del sistema fue un servidor Asterisk versión 1.8.16.0, debido a que la versión 1.8 de Asterisk actualmente es una de las versiones estable de citada aplicación y la subversión 16.0 fue la que mostro el mejor desempeño en la fase de pruebas del servidor e igualmente fue la única de esta versión compatible con el códec AMR el cual trabaja de manera nativa en los dispositivos móviles Blackberry.

Como ya se mencionó con anterioridad la finalidad de este trabajo es plantear un sistema de voz IP empleando una red inalámbrica basada en tecnología WiFi (*Wireless Fidelity*) con el fin de explotar las bondades de los dispositivos móviles.

A la fecha de realización de este trabajo existe una gran evolución de los dispositivos móviles Blackberry, por lo que la nueva versión de ese sistema operativo presenta algunas herramientas para el desarrollo de aplicaciones compatibles con el sistema operativo Android, esperándose que en próximas fechas las aplicaciones disponibles para el sistema operativo Android sean compatibles para la plataforma Blackberry, entre ellas, por supuesto las aplicaciones de voz IP.



**Figura 3.1 Diagrama del modelo propuesto.**

En la figura 3.1 se presenta un diagrama del modelo propuesto para la implementación de servicio de voz a dispositivos móviles, el cual consta de un servidor de voz, quien realizara la función de registro de usuarios y enlace de los usuarios para el establecimiento de la comunicación, para ello se tienen los requerimientos que se enlistan en los párrafos siguientes.

### **3.2 Servidor de voz.**

Para el dimensionamiento de las características es necesarios primeramente realizar un estudio de las necesidades de comunicación que va a proporcionar el sistema.

No existe una manera de medir el rendimiento de un sistema de voz IP, esto depende las características con las que se requiera implementar el sistema.

Una regla para calcular los requerimientos del sistema es que a mayor cantidad de usuarios y principalmente considerando el número de llamadas cursadas de manera simultánea, se necesita mayor velocidad de procesamiento y capacidad de almacenamiento del servidor de voz.

El servidor de voz puede ser instalado desde un equipo de cómputo con capacidades y características limitadas, hasta un servidor con características técnicas de vanguardia, el único y principal requisito es que cuenta con tarjeta de red.

En este trabajo se realizaron pruebas de funcionamiento con dos equipos diferentes:

1. Equipo de escritorio marca HP
2. Servidor marca HP, modelo PROLIANT, características:
  - 3 procesadores Intel Cuad-Core Xeon a 2.4 GHz.
  - Memoria cache 12288 KB.
  - Disco Duro 500 GBits.
  - Plataforma de 64 bits.

El Conmutador Telefónico (PBX), con el que se realizaron pruebas de funcionamiento es la versión 1.8.16.0 de Asterisk.

La aplicación Asterisk se encuentra desarrollada para el sistema operativo Linux, por lo que es necesario que nuestro servidor de voz cuente con citado sistema operativo, para el caso de estudio se instaló la distribución de RedHat CentOS 6.3.

### **3.2.1 Instalación del sistema operativo.**

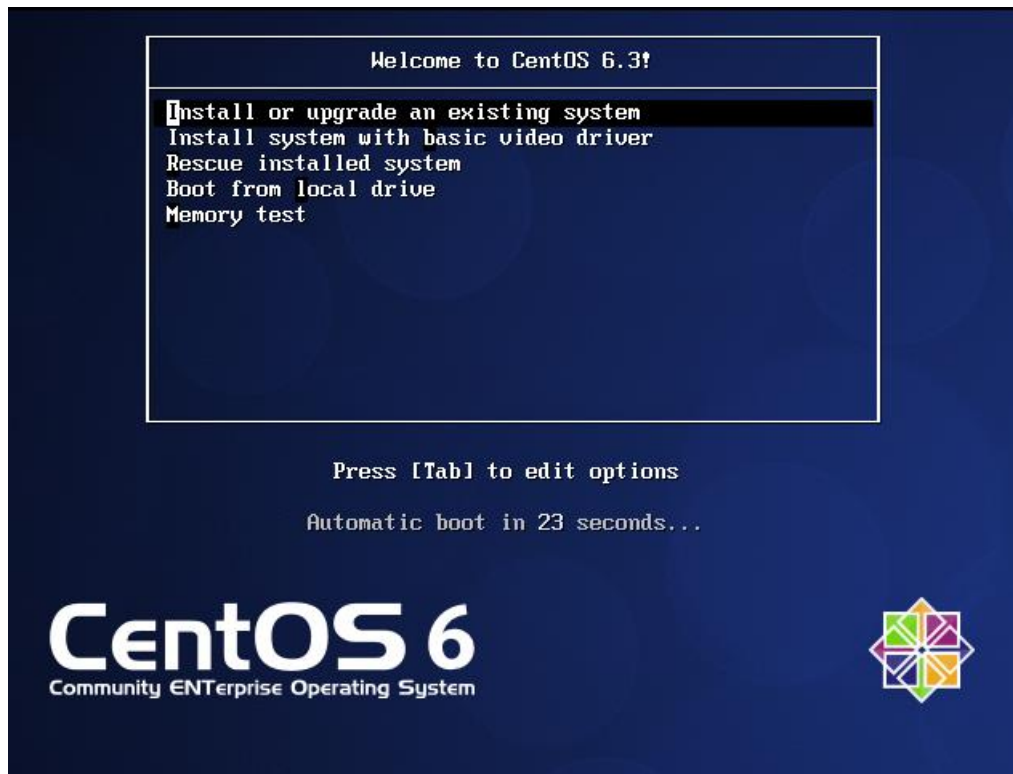
#### **Requerimientos**

- Se descarga el CD de Instalación de CentOS 6.3.

#### **Instalación**

Se arranca el sistema desde la unidad de DVD.





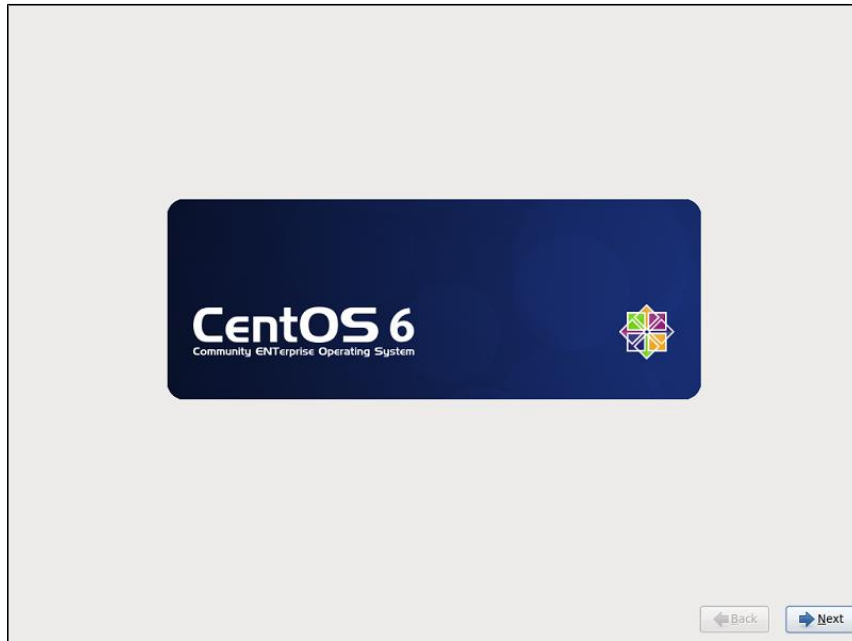
**Figura 3.2 Inicio de instalación de CentOS 6.**

Nos pregunta si queremos comprobar el disco duro. Seleccionamos ok.



**Figura 3.3 Comprobación del disco duro en CentOS 6.**

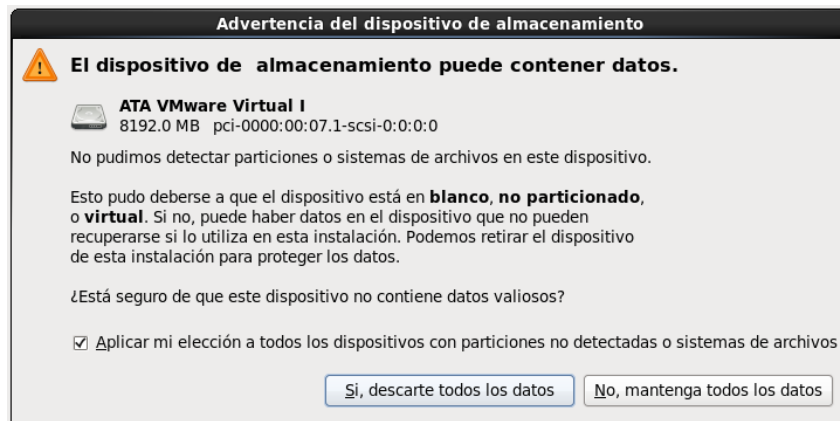
A partir de este momento la instalación es muy intuitiva y fácil de realizar por, únicamente es necesario seguir las instrucciones que van apareciendo en el proceso de instalación.



**Figura 3.4 Inicio de configuración de CentOS 6.**

Se selecciona el idioma de instalación, distribución del teclado y opción de dispositivos de almacenamiento básicos que se requieran.

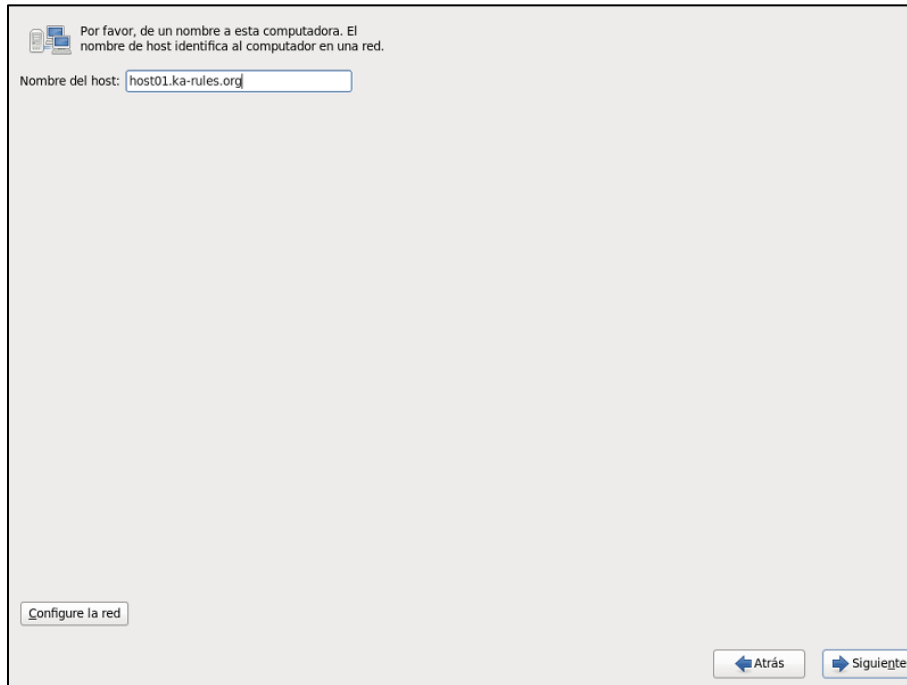
A continuación se observa una ventana de advertencia que nos indica si queremos eliminar los datos del disco o mantener los datos intactos.



**Figura 3.5 Advertencia del dispositivo de almacenamiento en CentOS 6.**

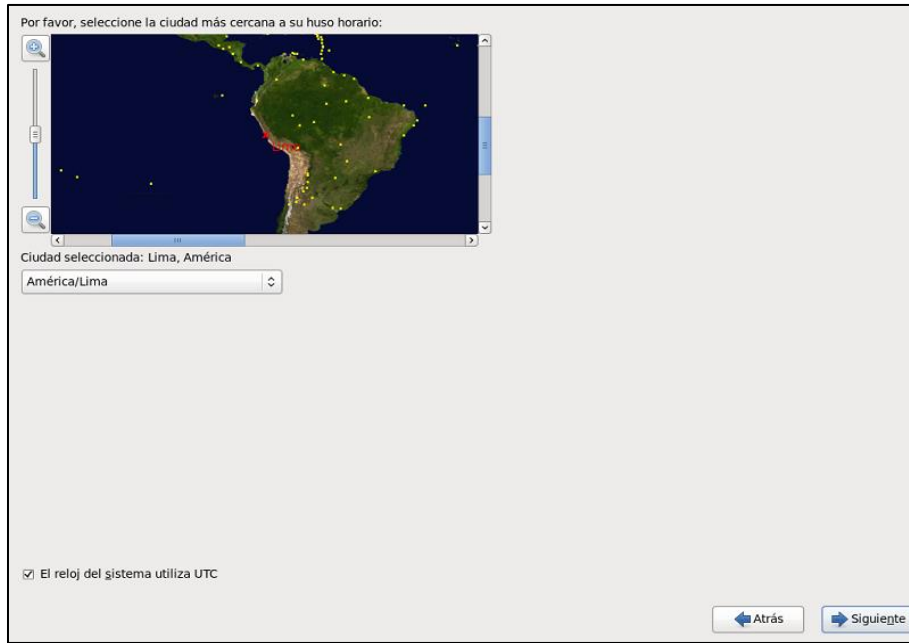
Seleccionar “Si, descarte todos los datos” para eliminar los datos almacenados en los dispositivos de almacenamiento del sistema.

Asignar el nombre de host para nuestro sistema



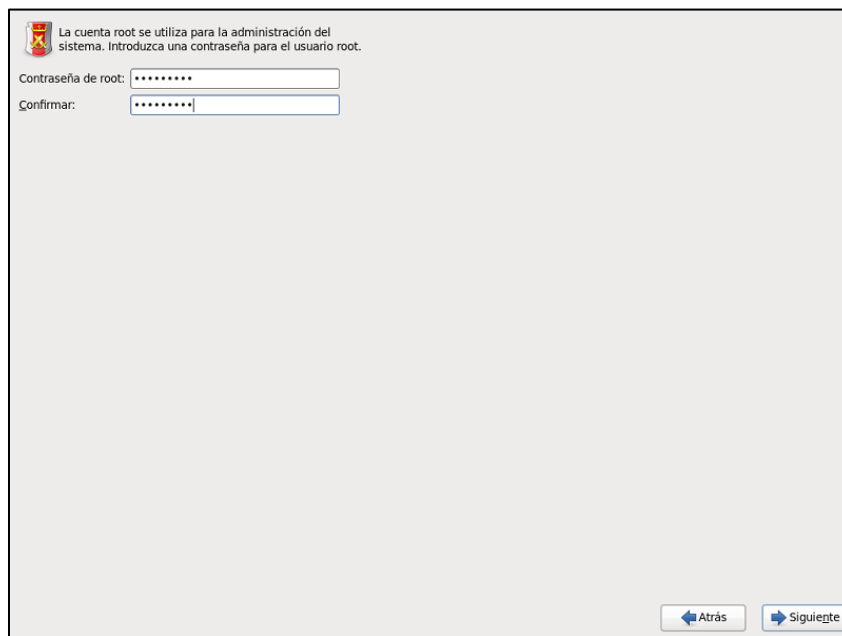
**Figura 3.6 Asignación del host en CentOS 6.**

Selección de la zona horaria y ubicación.



**Figura 3.7 Selección de zona horaria y ubicación en CentOS 6.**

Se asigna la contraseña para la cuenta de súper usuario (*root*).



**Figura 3.8 Selección de contraseña de root en CentOS 6.**

Este parámetro es muy importante para el inicio de la sesión en modo root para la configuración de la central telefónica.

Selección del tipo de instalación reemplazando las particiones existentes.

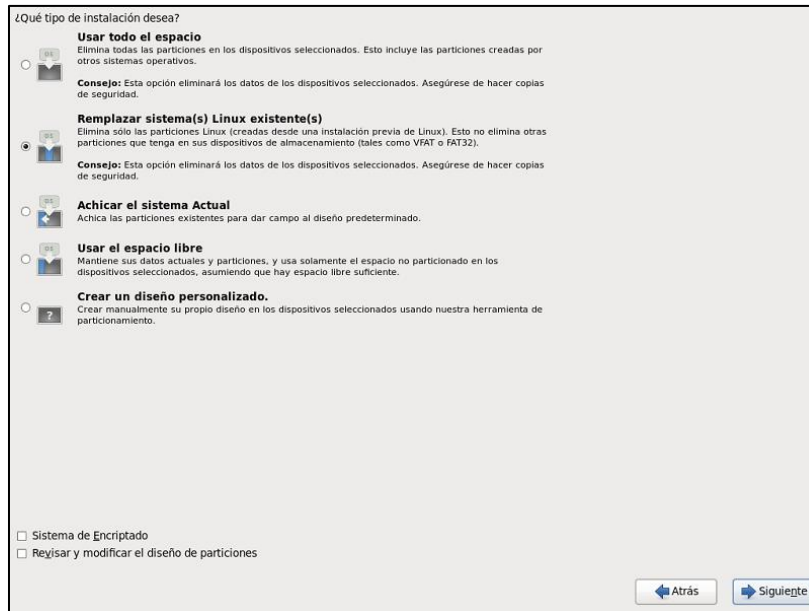
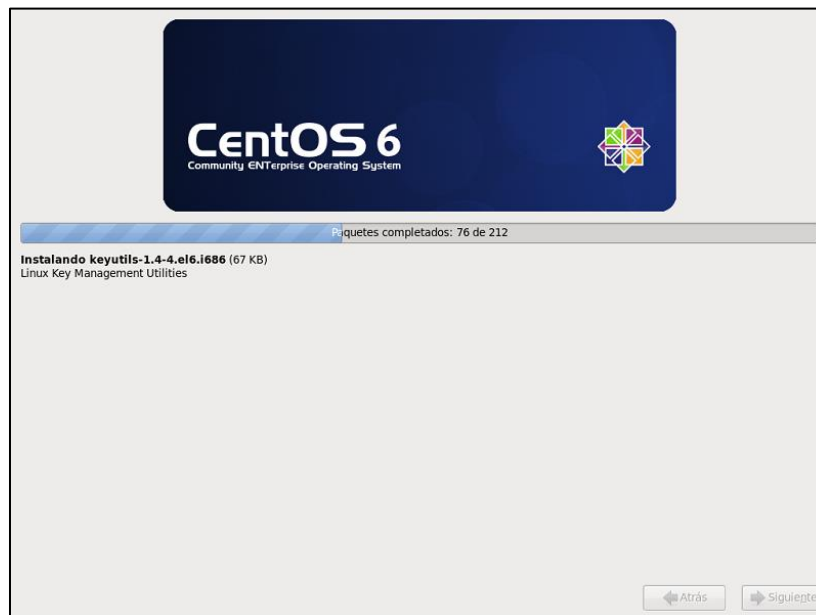


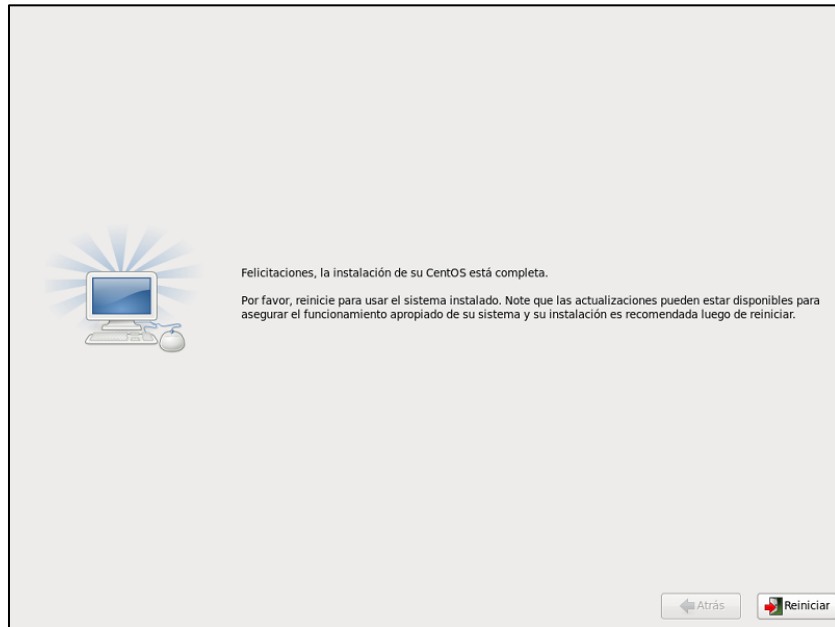
Figura 3.9 Selección del tipo de instalación en CentOS 6.

Una vez configurado estos parámetros inicia la instalación del sistema operativo.



**Figura 3.10 Instalación de CentOS 6.**

Al completar la instalación, se observa una ventana informando que se instaló el sistema con éxito, por lo que se debe reiniciar el sistema.



**Figura 3.11 Finalización de la instalación de CentOS 6.**

Después de reiniciar el sistema, este se encuentra listo para usarse.

```
CentOS release 6.3 (Final)
Kernel 2.6.32-279.el6.i686 on an i686

host01 login: _
```

**Figura 3.12 Registro para inicio de sesión en modo consola para CentOS 6.**

## **3.2.2 Instalación y configuración de la central telefónica.**

### **3.2.2.1 Configuración del sistema**

Antes de iniciar la instalación de la central telefónica, se necesita preparar el sistema.

Para la configuración de la tarjeta de red, se ingresa al archivo de configuración de la tarjeta de red, para ello es necesario comprobar las interfaces o tarjetas de red con las que cuenta el servidor, para ello se ejecuta el siguiente procedimiento en la consola del servidor:

Se ingresa en la carpeta donde se encuentran los archivos de configuración de las interfaces de red del equipo:

```
# cd /etc/sysconfig/network-scripts
```

Dependiendo la interfaz de red que se vaya a emplear, se edita el archivo de configuración de la interfaz seleccionada, esto depende de las tarjetas de red con que cuenta el servidor, en este caso se selecciona la interfaz eth0, por lo que para editarla se ejecuta la siguiente instrucción:

```
# vi ifcfg-eth0
```

Se configuran los parámetros como sigue:

```
DEVICE=eth0
BOOTPROTO= none
ONBOOT=yes
TYPE=Ethernet
IPADDR=192.168.0.1
NETMASK=255.255.255.0
DNS1=148.204.103.2
DNS2=148.204.102.3
```

Se instala la paquetería adicional del sistema operativo necesaria para la instalación de la central telefónica Asterisk.

```
# yum install -y nano gcc ncurses-devel libtermcap-devel
kernel-devel gcc-c++ openssl-devel newt-
devel zlib-devel unixODBC-devel libtool
libtool-devel make wget libxml2-devel
pkgconfig autoconf automake unzip patch
```

Instalación de las cabeceras de Linux

```
# yum install -y install kernel kernel-devel kernel-headers
```

Reiniciar el sistema para aplicar los cambios.

```
# reboot
```

### 3.2.2.2 Instalación de la central telefónica.

Se descarga la aplicación de la central telefónica Asterisk, en la carpeta /usr/src

```
# cd /usr/src
```

Descarga de la versión de asterisk que se va a instalar

```
# wget http://downloads.asterisk.org/pub/telephony/asterisk/  
old-releases/asterisk-1.8.16.0.tar.gz
```

Este archivo viene empaquetado en formato .tar, por lo que es necesario desempaquetarlo con la siguiente instrucción

```
# tar -xzf asterisk-1.8.16.0.tar.gz
```

Al ejecutar este comando se crea un nuevo directorio con el nombre de la versión que se descargo, en este caso asterisk-1.8.16.0, en este directorio se descargan las librerías necesarias para la compilación de la central telefónica Asterisk.

Se ingresa al directorio asterisk-1.8.16.0

```
# cd /usr/src/asterisk-1.8.16.0
```

Para iniciar la compilación e instalación de la central telefónica, se realiza el siguiente procedimiento.

```
# ./configure
```

Con este comando se revisan las dependencias que se encuentran instaladas en el sistema.

```
# make menuselect
```



Se seleccionan los módulos a instalar, para la instalación inicial se recomienda cargar los módulos que se encuentran seleccionados por default

Se salvan los cambios y se sale del menú de selección de módulos.

```
# make
```

Compilamos

```
# make install
```

Creamos los scripts de autoarranque

```
# make config
```

Cargamos los archivos de muestra

```
# make samples
```

Esta última instrucción no deberá ser ejecutada cuando ya contamos con un entorno en desarrollo, es decir cuando ya tenemos una central telefónica en funcionamiento, ya que de ejecutarlos se pierde la configuración realizada en el archivo del plan de marcación (*extensions.conf*) y en el archivo de configuración de usuarios (*sip.conf*).

Se edita el archivo de configuración de arranque de Linux, con el fin de configurar la inicialización de la Interface de Línea de Comandos (CLI) de asterisk

```
# nano /etc/selinux/config
```

Cambiamos el siguiente parámetro

```
SELINUX=enforcing
```

por

```
SELINUX= disabled
```

Con el fin de facilitar la edición de los archivos de configuración de la central telefónica, es recomendable descargar la sintaxis de colores del editor nano, por lo que se ejecuta la siguiente instrucción:

```
# wget -O /usr/share/nano/asterisk.nanorc
      http://asteriskmx.com/t/asterisk.nanorc
```

Con esta instrucción se descarga el módulo ubicado en la dirección <http://asteriskmx.com/t/asterisk.nanorc> en la ruta /usr/share/asterisk.nanorc del servidor.

Se edita el archivo nanorc

```
# nano /etc/nanorc
```

Agregamos la siguiente línea al final del archivo

```
include "/usr/share/nano/asterisk.nanorc"
```

Para subir el nivel de verbosidad, se edita el archivo asterisk.conf, este parámetro se configura para obtener mayor información en nuestra Interfaz de Línea de Comandos (*CLI Command Line Interface*), necesario para la detección y corrección de errores en el sistema.

```
# nano /etc/asterisk/asterisk.conf
```

Se modifica el nivel de verbosidad a 3, quedando de la siguiente manera:

```
[options]
verbose=3
```

Se inicia nuevamente del servicio de asterisk

```
#!/etc/init.d/asterisk start
```

Se reconecta a la consola de asterisk

```
# asterisk -r
```

En pantalla aparece la siguiente línea:

```
asterisk*CLI>
```

Para salir nuevamente de la Interfaz de Línea de Comandos se ejecuta la instrucción:

```
CLI>exit
```

### 3.2.2.3 Instalación del códec AMR-NB

Para iniciar la instalación del códec AMR-NB (*Adaptive Multi-Rate Narrow Band*) RFC 4867 [6] es necesario detener el servicio de asterisk mediante la siguiente instrucción

```
# service asterisk stop
```

Se ingresa a la ruta /usr/src para la descarga del modulo

```
#cd /usr/src
```

Se descarga el parche del codec AMR para la central telefónica que se tiene instalada en el servidor

```
#wget http://sourceforge.net/projects/asterisk-amr/files/  
1.8.0-rc2_asterisk_amr_patch.diff/download
```

Considere que en la instrucción anterior después de la ruta que se descarga el parche, se debe escribir sin espacios, de igual manera el parche que se descarga está diseñado para la versión 1.8.0-rc2 de asterisk, el cual es compatible con la versión 1.8.16.0, esta se empleó para el desarrollo del presente trabajo ya que la versión 1.8.0-rc2 es un versión rc (*release candidate*) que fue liberada pero que cuenta con fallas y por lo tanto no es estable.

Se aplica el parche a la versión de asterisk

```
# cd asterisk-1.8.16.1/ && patch -p2 < ../1.8.0-  
rc2_asterisk_amr_patch.diff
```

Se direcciona a la ubicación asterisk-1.8.16.1/codecs/amr

```
#cd /usr/src/asterisk-1.8.16.1/codecs/amr
```

Se descarga el codec AMR.

```
#wget http://www.3gpp.org/ftp/Specs/archive/26\_series/26.104/26104-700.zip
```

Se descomprime el archivo 26104-700.zip

```
#unzip 26104-700.zip
```

Al descomprimir el archivo se genera el archivo 26104-700\_ANSI\_C\_source\_code.zip, el cual es necesario descomprimir

```
#unzip -j 26104-700_ANSI_C_source_code.zip
```

Se ingresa a la ruta us/src/asterisk-1.8.16.0

```
#cd /usr/src/asterisk-1.8.16.0
```

Se compila e instala nuevamente la central telefónica asterisk con el módulo AMR que de ha configurado

```
#!/configure  
#make  
#make install
```

Se reinicia el servicio de asterisk

```
#service asterisk start
```

Se reinicia la consola de asterisk

```
#asterisk -r
```

Para comprobar que el módulo AMR se encuentra cargado, se ejecuta la siguiente instrucción:

```
*CLI>core show codecs
```

```
localhost*CLI> core show codecs
Disclaimer: this command is for informational purposes only.
It does not indicate anything about your configuration.
```

INT	BINARY	HEX	TYPE	NAME	DESCRIPTION
1	(1 << 0)	(0x1)	audio	g723	(G.723.1)
2	(1 << 1)	(0x2)	audio	gsm	(GSM)
4	(1 << 2)	(0x4)	audio	ulaw	(G.711 u-law)
8	(1 << 3)	(0x8)	audio	alaw	(G.711 A-law)
16	(1 << 4)	(0x10)	audio	g726aal2	(G.726 AAL2)
32	(1 << 5)	(0x20)	audio	adpcm	(ADPCM)
64	(1 << 6)	(0x40)	audio	slin	(16 bit Signed Linear PCM)
128	(1 << 7)	(0x80)	audio	lpc10	(LPC10)
256	(1 << 8)	(0x100)	audio	g729	(G.729A)
512	(1 << 9)	(0x200)	audio	speex	(SpeeX)
1024	(1 << 10)	(0x400)	audio	ilbc	(iLBC)
2048	(1 << 11)	(0x800)	audio	g726	(G.726 RFC3551)
4096	(1 << 12)	(0x1000)	audio	g722	(G722)
8192	(1 << 13)	(0x2000)	audio	amr	(AMR NB)
16384	(1 << 14)	(0x4000)	audio	siren14	(ITU G.722.1 Annex C, (Siren14, licensed from Polycom))
32768	(1 << 15)	(0x8000)	audio	slin16	(16 bit Signed Linear PCM (16kHz))
65536	(1 << 16)	(0x10000)	image	jpeg	(JPEG image)
131072	(1 << 17)	(0x20000)	image	png	(PNG image)
262144	(1 << 18)	(0x40000)	video	h261	(H.261 Video)
524288	(1 << 19)	(0x80000)	video	h263	(H.263 Video)
1048576	(1 << 20)	(0x100000)	video	h263p	(H.263+ Video)
2097152	(1 << 21)	(0x200000)	video	h264	(H.264 Video)
4194304	(1 << 22)	(0x400000)	video	mpeg4	(MPEG4 Video)
8388608	(1 << 23)	(0x800000)	video	unknown	(unknown)
16777216	(1 << 24)	(0x1000000)	video	unknown	(unknown)

**Figura 3.13** Codecs instalados en la central telefónica.

La figura 3.12 muestra los codecs instalados en la central telefónica, se puede observar en el recuadro color rojo que ya se encuentra instalado el codec AMR, de igual manera es necesario comprobar que el códec AMR se encuentra listo para su empleo, por lo que se comprueba que la central telefónica se encuentra lista pueda realizar el proceso para la conversión de codecs, es decir el transcoding, mediante la siguiente instrucción:

```
*CLI>core show translation
```

```
localhost*CLI> core show translation
Translation times between formats (in microseconds) for one second of data
Source Format (Rows) Destination Format (Columns)
```

	g723	gsm	ulaw	alaw	g726aal2	adpcm	slin	lpc10	g729	speex	ilbc	g726	g722	amr	siren14	slin16	g719	speex16	siren7	testlaw	
g723	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
gsm	-	-	2	2	2000	2	1	1001	-	-	-	1001	2	4000	-	1002	-	-	-	-	2
ulaw	-	2	-	1	2000	2	1	1001	-	-	-	1001	2	4000	-	1002	-	-	-	-	2
alaw	-	2	1	-	2000	2	1	1001	-	-	-	1001	2	4000	-	1002	-	-	-	-	2
g726aal2	-	2	2	2	-	2	1	1001	-	-	-	1001	2	4000	-	1002	-	-	-	-	2
adpcm	-	2	2	2	2000	-	1	1001	-	-	-	1001	2	4000	-	1002	-	-	-	-	2
slin	-	1	1	1	1999	1	-	1000	-	-	-	1000	1	3999	-	1001	-	-	-	-	1
lpc10	-	2	2	2	2000	2	1	-	-	-	-	1001	2	4000	-	1002	-	-	-	-	2
g729	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
speex	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
ilbc	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
g726	-	1001	1001	1001	2999	1001	1000	2000	-	-	-	-	1001	4999	-	2001	-	-	-	-	1001
g722	-	2	2	2	2000	2	1	1001	-	-	-	1001	-	4000	-	1000	-	-	-	-	2
amr	-	1001	1001	1001	2999	1001	1000	2000	-	-	-	2000	1001	-	-	2001	-	-	-	-	1001
siren14	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
slin16	-	1002	1002	1002	3000	1002	1001	2001	-	-	-	2001	1000	5000	-	-	-	-	-	-	1002
g719	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
speex16	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
siren7	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
testlaw	-	2	2	2	2000	2	1	1001	-	-	-	1001	2	4000	-	1002	-	-	-	-	-

Figura 3.14 Tiempo para la realización de Transcoding en asterisk 1.8.16.0.

En la figura 3.13 las franjas de color rojo muestran el tiempo en microsegundo en que el sistema realiza la transcoding entre el codec AMR y los diversos codecs instalados en el servidor, se observa que en la columna y el renglón de AMR existen valores numéricos con los codecs que instalados, por ejemplo, al observar la columna del codec siren14, no se visualiza ningún valor numérico únicamente se observan guiones (-), esto indica que el modulo del codec siren14 no se encuentra cargado en el sistema.

Se sale de la consola de asterisk

```
*CLI>exit
```

Se reinicia el sistema para configurar los cambios

```
#reboot
```

Una vez que se instaló el codec AMR es necesario configurar este para su empleo en redes móviles, mediante este proceso se reduce el ancho de banda para optimizar su empleo, para lo anterior se edita el archivo codecs.conf

```
#nano /etc/asterisk/codecs.conf
```

Al final del archivo se agregan las siguientes líneas de código

```
[amr]
octet-aligned=1
mode=MR475
dtx=0
cng=1
vad=0
```

Se salvan los cambios al archivo

```
Ctrl+O
```

Y se sale del archivo

```
Ctrl+X
```

Se carga el modulo

```
#asterisk -r
#module reload codec_amr.so
```

Y se reinicia nuevamente el sistema

```
#reboot
```

Una vez que se han configurado satisfactoriamente el codec AMR y la central telefónica, es necesaria la creación de un plan de marcación y el alta de los usuarios.

Se edita el archivo `extensions.conf` ubicado en el directorio `/etc/asterisk`

```
#nano /etc/asterisk/extensions.conf
```

Este archivo contiene las instrucciones ejemplo del plan de marcaciones debido que fueron instaladas en la configuración inicial, existen diferentes formas de crear el plan de marcación, estas dependen del estilo propio del administrador del sistema, en este trabajo se creó el plan de marcación que se muestra a continuación:

```
[general]
static=yes
writeprotect=no
clearglobalvars=no

[globals]
CONSOLE=Console/dsp

[internas]
exten => 111,1,Playback(demo-congrats)
exten => _1xx,1,Dial(SIP/${EXTEN})
exten => _1xx,n,hangup
```

En la primera parte se creó el campo `[general]`, en el cual se agrupan las características generales del plan de marcación y serán aplicados a todo el plan de marcación.

En el campo `[globals]` se crean las variables, estas van a ser capturadas por la consola de asterisk por medio de los valores ingresados al sistema por medio del teclado.

El campo `[internas]` crear el contexto, en este se definen las acciones a ejecutar en el plan de marcación, en este caso al ingresar el número `111` se buscar el archivo de audio llamado `demo-congrats` y se reproduce el archivo mediante la instrucción `Playback()`.

Para que los cambios surtan efectos es necesario recargar el plan de marcación

```
*CLI>dialplan reload
```

Para finalizar la configuración de la central telefónica es necesario crear los usuarios que van a emplear el sistema, para ello se edita el archivo `sip.conf` del sistemas en la ruta `/etc/asterisk/sip.conf`

```
#nano /etc/asterisk/sip.conf
```



Se realizan los cambios en el archivo, quedando de la manera siguiente:

```
[general]
timert1=3000
rtptimeout=15
jbenable=yes
jbforce=no
jbmaxsize=200
jbresyncthreshold=1200
jbimpl=fixed
```

```
[usuario]
context=internas
canreinvite=no
type=friend
host=dynamic
allow=all
nat=no
qualify=yes
```

```
[100](usuario)
secret=100
```

```
[101](usuario)
secret=101
```

```
.
.
.
.
```

```
[110](usuario)
Secret=110
```

## Capítulo 4

# RESULTADOS.

### 4.1 Registro de usuarios.

El primer paso para el establecimiento de una llamada es el registro de los usuarios, el requerimiento para el registro de usuarios es que los usuarios se encuentren registrados en la red, que tenga asignada una dirección IP no necesariamente con una dirección IP fija, el router puede estar asignando una dirección IP a través del protocolo de configuración dinámica de Host ( *Dynamic Host Configuration Protocol DHCP*).

```
-- Registered SIP '100' at 148.204.34.106:53656  
-- Registered SIP '101' at 148.204.34.106:51436
```

Figura 4.1 Registro de usuarios.

En la figura 4.1 se muestra el mensaje de consola al momento que dos usuarios se registran al servidor, este mensaje nos proporciona la siguiente información:

- Registered SIP '100'

Muestra el registro de un usuario SIP con la identificación '100'

- 148.204.34.106:53516

Nos indica la dirección IP (148.204.34.106) asignada al usuario que se está registrando y el puerto (53516) al que se está conectando.

En el caso de la Figura 4.1 nos muestra que los usuarios 100 y 101 se encuentran registrados al sistema, aunque se registran de la misma dirección IP, el puerto es diferente, en este caso en específico los dispositivos móviles se deben registrar a la dirección IP 148.204.35.165, nuestro dispositivo móvil con el username '100' tiene asignada la dirección IP 192.168.0.100 la cual fue asignada de manera dinámica por el ruteador, la dirección IP del servidor es una IP fija por lo tanto puede ser vista desde cualquier lugar de la internet, la dirección asignada por el ruteador a nuestro usuario es una dirección de la red que el propio ruteador está creando, para que nuestro usuario pueda ser ubicado desde otra red, por ejemplo la internet es necesario el empleo del 'NATEO', este término se deriva 'NAT' (*Name Address Translation*) mediante este proceso podemos identificar a un usuario dentro de una red de área local en una red de Área Metropolitana (MAN) o de Area Amplia (WAN), es por ello que no vemos la dirección IP asignada por el ruteador al dispositivo, sino la IP asignada por el ruteador para que pueda ser ubicado por un usuario externo a la red local.

Para comprobar los usuarios registrados en nuestro servidor empleamos la siguiente instrucción en la Interfaz de Línea de Comandos de asterisk (*CLI*)

```
localhost*CLI> sip show peers
```

```
localhost*CLI> sip show peers
Name/username      Host                Dyn Forcerport ACL Port      Status
100/100            148.204.34.106     D N          53516      OK (51 ms)
101/101            148.204.34.106     D N          53720      OK (28 ms)
102                (Unspecified)      D N          0          UNKNOWN
103/103            (Unspecified)      D N          0          UNKNOWN
104/104            (Unspecified)      D N          0          UNKNOWN
105/105            (Unspecified)      D N          0          UNKNOWN
106                (Unspecified)      D N          0          UNKNOWN
107                (Unspecified)      D N          0          UNKNOWN
108                (Unspecified)      D N          0          UNKNOWN
109                (Unspecified)      D N          0          UNKNOWN
110/110            (Unspecified)      D N          0          UNKNOWN
```

Figura 4.2 Usuarios registrador en el servidor

En el área sombreada color rojo en la Figura 4.2 se muestra que el usuario '100' se ha registrado al servidor, el marcado con rectángulo color azul muestra que el usuario '101' de

igual manera se ha registrado al sistema, no así los demás usuarios, los cuales no se han registrado al servidor, ya que en la columna Host no se cuenta con una dirección IP válida, por el contrario se muestra el término (*Unspecified*).

Los usuarios '100' y '101' se encuentran listos para el establecimiento de una llamada.

Mediante la instrucción

```
localhost*CLI> sip show peer <usuario>
```

Podemos verificar las características de un usuario específico

```

localhost*CLI> sip show peer 100

* Name      : 100
Secret     : <Set>
MD5Secret  : <Not set>
Remote Secret: <Not set>
Context    : internas
Subscr.Cont. : <Not set>
Language   :
AMA flags  : Unknown
Transfer mode: open
CallingPres : Presentation Allowed, Not Screened
Callgroup  :
Pickupgroup :
MOH Suggest :
Mailbox    :
VM Extension : asterisk
LastMsgsSent : 32767/65535
Call limit : 0
Max forwards : 0
Dynamic    : Yes
Callerid   : "" <>
MaxCallBR  : 384 kbps
Expire     : 21
Insecure   : no
Force rport : Yes
ACL        : No
-----
DirectMedACL : No
T.38 support : No
T.38 EC mode : Unknown
T.38 MaxDtgrm: -1
DirectMedia  : No
PromiscRedir : No
User=Phone   : No
Video Support: No
Text Support : No
Ign SDP ver  : No
Trust RPID   : No
Send RPID    : No
Subscriptions: Yes
Overlap dial : Yes
DTMFmode     : rfc2833
Timer T1     : 3000
Timer B      : 192000
ToHost       :
Addr->IP     : 148.204.34.106:51340
Defaddr->IP  : (null)
Prim.Transp. : UDP
Allowed.Trsp : UDP
Def. Username: 100
SIP Options  : (none)
Codecs       : 0x80430c7fffff (g723|gsm|ulaw|alaw|g726|adpcm|slin|lpc10|g729|speex|speex16|ilbc|g726aal2|g722|amr|slin|
p|h264|mpeg4|red|t140|siren7|siren14|testlaw|g719)
Codec Order  : (none)
Auto-Framing : No
100 on REG   : No
Status       : OK (50 ms)
Useragent    : BlackberryVoipDialer
Reg. Contact : sip:100@192.168.0.100:51340
Qualify Freq : 60000 ms
Sess-Timers  : Refuse
Sess-Refresh : uas
Sess-Expires : 1800 secs
Min-Sess     : 90 secs
RTP Engine   : asterisk
Parkinglot   :
Use Reason   : No

```

Figura 4.3 Características de un usuario específico.

En la figura 4.3 se muestran las características del usuario 100, la instrucción nos despliega los parámetros generales del usuario que en ese momento se encuentra registrado al sistema, el análisis de estos parámetros nos proporciona información como la dirección IP

asignada al equipo, el dispositivo que se seta empleando, el tiempo de registro, tipo de codecs que pueden ser empleados, entre otros muchos otros mas.

Existen algunas otras instrucciones que pueden ser empleadas en el sistema, las cuales nos pueden presentar información sobre el establecimiento de las sesiones, entre ellos el siguiente:

```
localhost*CLI> sip set debug on
```

```
--- (11 headers 0 lines) ---
Sending to 201.144.162.4:22681 (no NAT)
<--- Transmitting (NAT) to 201.144.162.4:22681 --->
SIP/2.0 401 Unauthorized
Via: SIP/2.0/UDP 10.213.234.70:5060;branch=z9hG4bK863514722;received=201.144.162.4;rport=22681
From: <sip:156@148.204.35.165>;tag=354055244
To: <sip:156@148.204.35.165>;tag=as69bc1910
Call-ID: 592614938
CSeq: 10 REGISTER
Server: Asterisk PBX 1.8.0-rc2
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, SUBSCRIBE, NOTIFY, INFO, PUBLISH
Supported: replaces
WWW-Authenticate: Digest algorithm=MD5, realm="asterisk", nonce="60a90224"
Content-Length: 0
```

Figura 4.4 Instrucción sip set debug on.

## 4.2 Establecimiento de llamadas.

Una vez que se cuentan con usuarios registrados en el servidor de voz es posible establecer llamadas entre estos usuarios.

```
== Using SIP RTP CoS mark 5
-- Executing [101@internas:1] Dial("SIP/100-0000006a", "SIP/101") in new stack
== Using SIP RTP CoS mark 5
-- Called 101
-- SIP/101-0000006b is ringing
-- SIP/101-0000006b is ringing
-- SIP/101-0000006b answered SIP/100-0000006a
-- Locally bridging SIP/100-0000006a and SIP/101-0000006b
== Spawn extension (internas, 101, 1) exited non-zero on 'SIP/100-0000006a'
```

Figura 4.5 Proceso de una llamada.

En la figura 4.5 se muestra el proceso de una llamada , en el rectángulo sombreado en color amarillo se muestra el inicio de una llamada, iniciada por el usuario '101' hacia el usuario '100', se muestran los siguientes parámetros:

- == Using SIP RTP CoS mark 5  
-- Executing [101@internas:1] Dial("SIP/100-0000006a",  
"SIP/100") in new stack
- == Using SIP RTP CoS mark 5  
-- Called 101  
-- SIP/101-0000006b is ringing  
-- SIP/101-0000006b is ringing

Hasta este momento únicamente hemos iniciado y empleado el Protocolo de Inicio de Sesión (*Session Initiation Protocol SIP*), es decir la señalización para el inicio establecimiento y finalización de la llamada, aun no se ha iniciado el flujo de tráfico de voz mediante el Protocolo de Tiempo Real (*Real Time Protocol RTP*)

En el rectángulo color verde se muestra el momento en que el usuario 100 contesta la llamada al usuario 101 apareciendo en consola la siguiente linea :

- -- SIP/101-0000006b answered SIP/100-0000006a

A partir de este momento inicia el flujo de datos voz RTP y por lo tanto la llamada como tal.

De igual manera existe la instrucción

```
localhost*CLI>rtp set debug on
```





### 4.3 Captura de paquetes.

Para el análisis de paquetes que son cursados en el servidor de voz se empleó la aplicación Wireshark 1.8.5, esta aplicación es un analizador de protocolos para el análisis y una poderosa herramienta para solucionar problemas en redes de comunicaciones.

Wireshark permite ver todo el tráfico que pasa a través de una red estableciendo la configuración en modo promiscuo (acepte todos los paquetes sin importar el destinatario), permite examinar paquetes en una red activa o mediante un archivo de captura salvado en un dispositivo de almacenamiento, dentro de sus características puede mostrar de forma gráfica el establecimiento de una sesión SIP, lo cual facilita el análisis de la información de sesión, de igual manera cuenta con una poderosa herramienta para la decodificación de comunicaciones de voz IP.

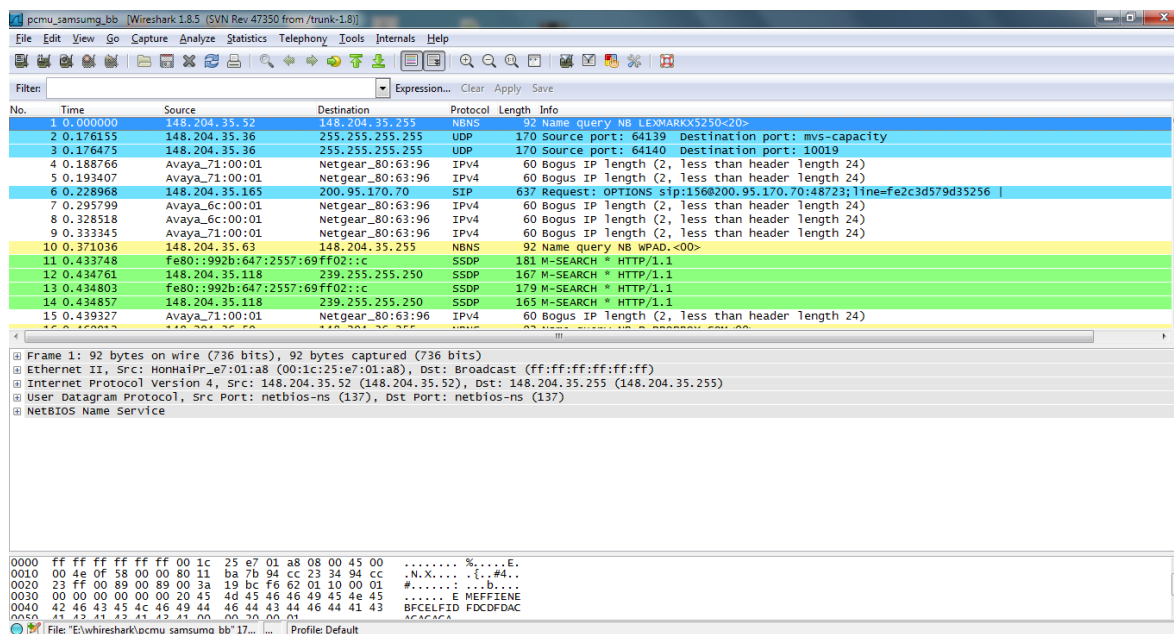


Figura 4.7 Interfaz gráfica del analizador de protocolos Whireshark.

En la figura 4.7 se muestra la interfaz gráfica del analizador de protocolos Whireshark al realizar una captura de paquetes, la interpretación y análisis completo de la información que nos proporciona esta aplicación quedan fuera del alcance de este trabajo, en este momento nos limitaremos únicamente a la captura de paquetes con protocolo SIP y RTP.

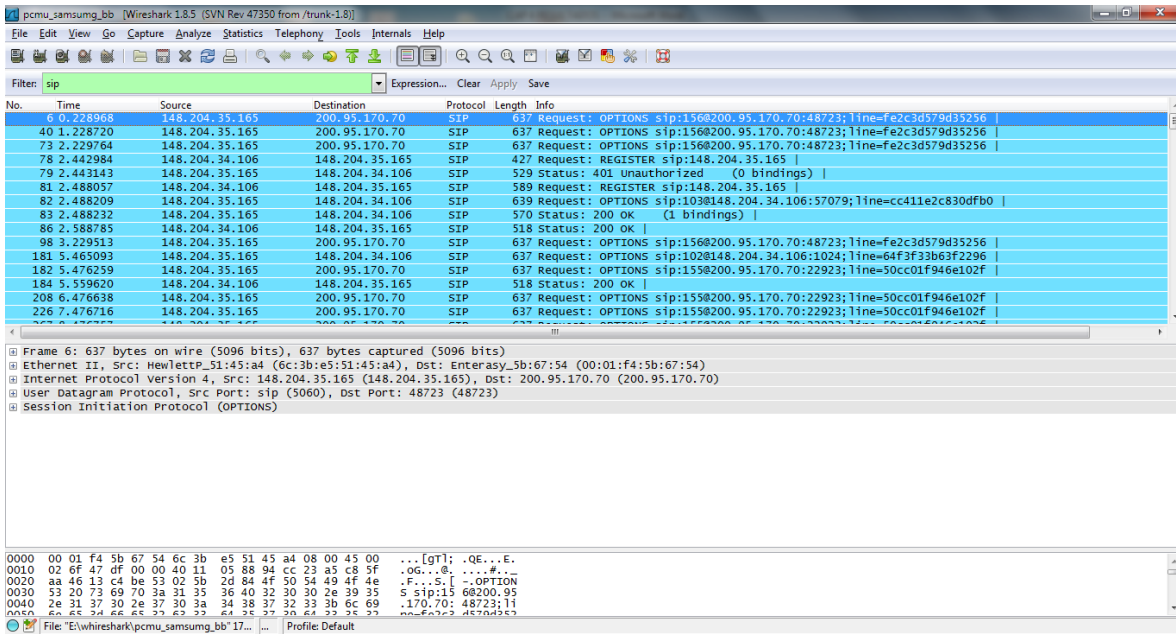


Figura 4.8 Filtrado de protocolo SIP en Whireshark.

En la figura 4.8 se muestra la interfaz gráfica de la aplicación Whireshark filtrando únicamente los paquetes que emplean el protocolo SIP, aunque existe la posibilidad de presentar este resultado de forma gráfica lo cual es más sencillo de interpretar

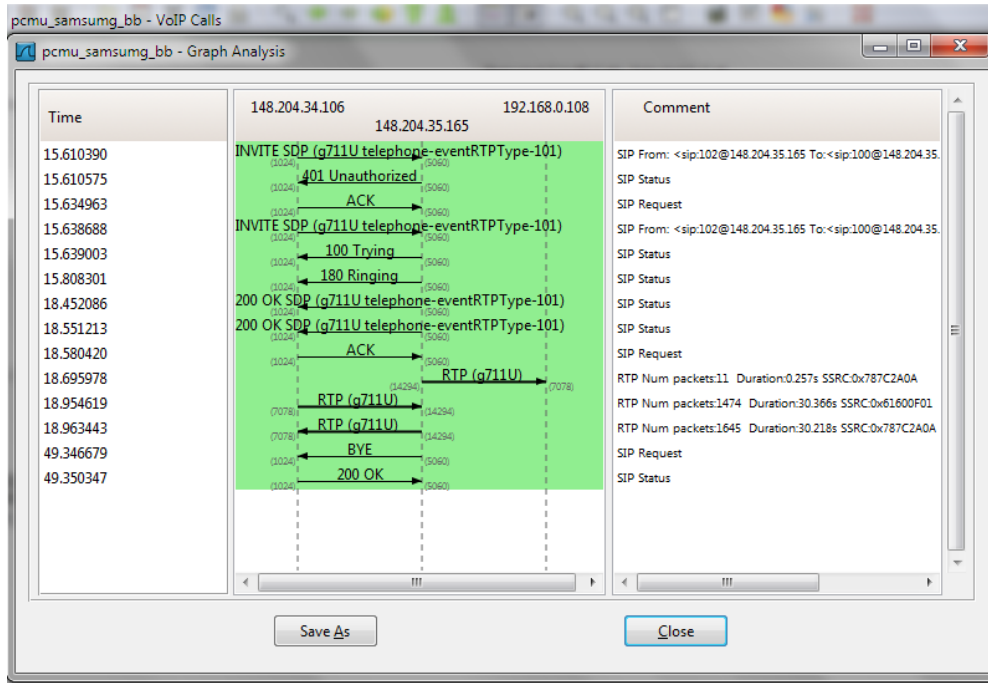


Figura 4.9 Flujo del protocolo SIP para el establecimiento de una sesión en Whireshark.

En la figura 4.9 se observa de manera gráfica el proceso de establecimiento de una llamada entre los usuarios ‘102’ y ‘100’, esta forma de análisis de la información es más intuitiva y fácil de interpretar, por lo que de ser necesario el análisis de información para detección y corrección de errores para el inicio de sesión es recomendable el empleo de este tipo de gráficas.

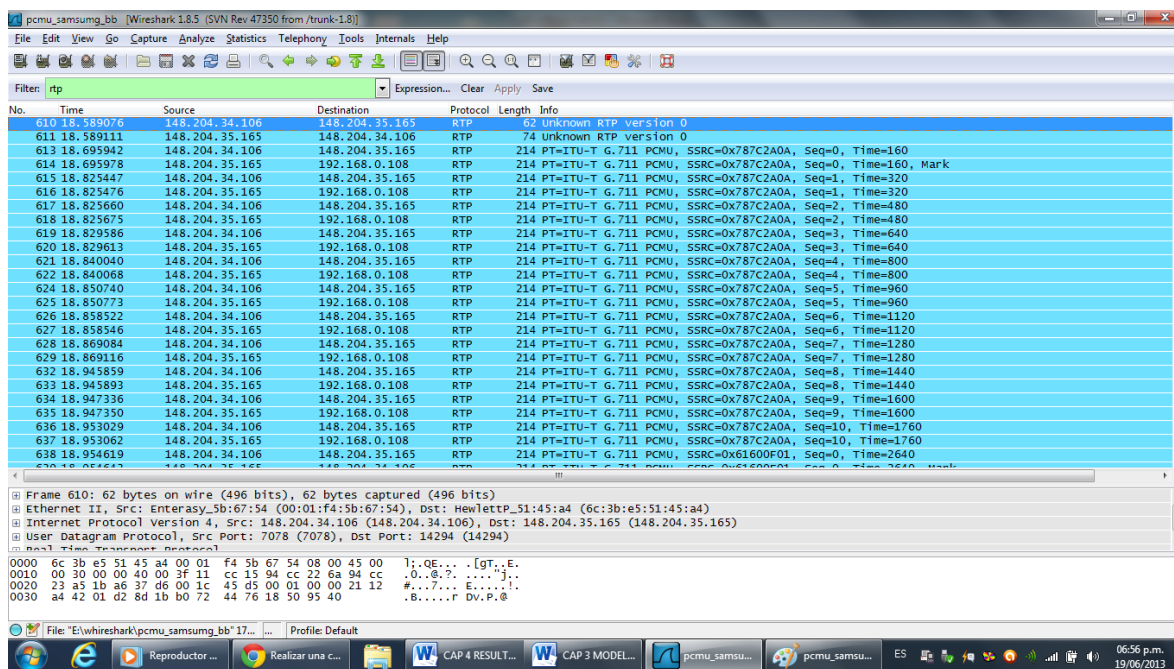


Figura 4.10 Captura de paquetes RTP en Whireshark.

En la figura 4.10 se muestra la captura de paquetes RTP durante es establecimiento de una llamada, en los primeros paquetes capturados podemos ver que se envía información sobre la versión del protocolo RTP que los usuarios están empleando, en este caso la versión 0, posteriormente podemos observar el flujo de paquetes entre ambos usuarios, en esta captura podemos obtener el codec que estamos empleando para el establecimiento de la llamada, en este caso el G-711 PCMU.

Whireshark cuenta con una herramienta para decodificar las llamadas, ya que se encuentran precargados los codecs para realizar este proceso, esto proporciona una ayuda para escuchar las llamadas que han sido capturadas por esta aplicación.

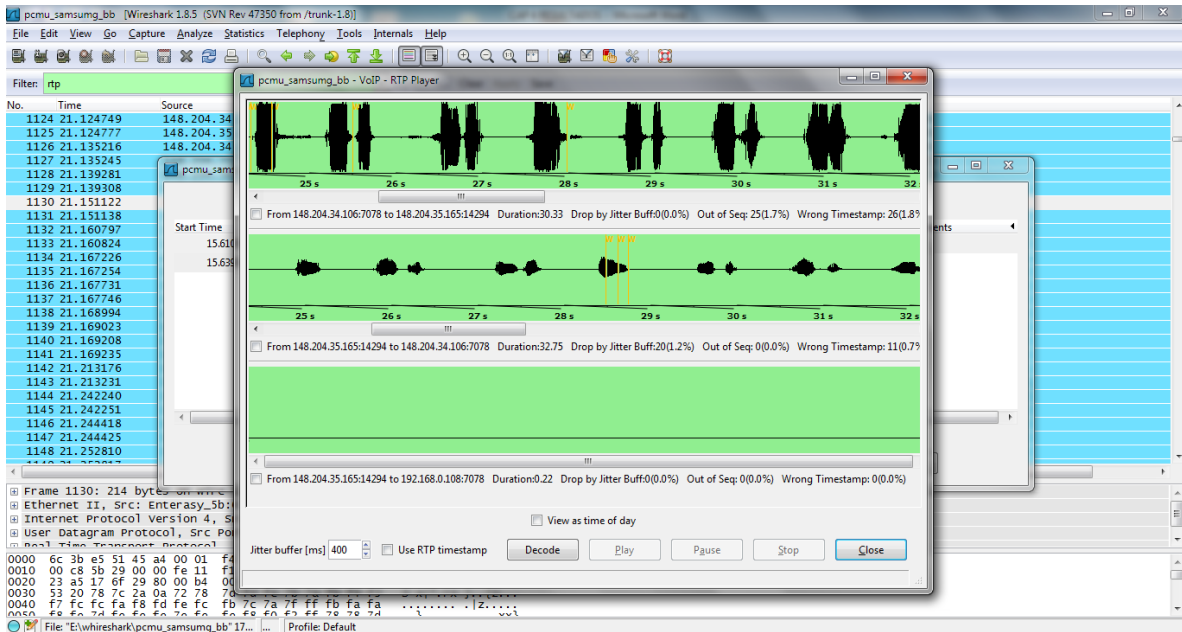


Figura 4.11 Decodificación de una llamada en Whireshark.

En la figura 4.11 se muestra la decodificación de los paquetes capturados por la aplicación Whireshark empleando el codec PCMU, en este proceso podemos escuchar la conversación entre ambos usuarios, en algunos casos esta aplicación es de gran utilidad siempre y cuando no se violen leyes de privacidad o se cuenten con las autorizaciones de los usuarios a fin de evitar responsabilidades legales que puedan provocar problemas legales.

### 4.3 Pruebas realizadas.

En las figuras 4.13 y 4.14 se muestran los resultados de una captura de paquetes con la aplicación Whireshark durante una llamada entre los usuarios ‘100’ (Blackberry 9900) y ‘103’ (iPhone) empleando el codec PCM ley A y ley  $\mu$ , al reproducir estos archivos tenemos voz audible y comprensible aunque con pérdidas de paquetes.

Usuario	Dispositivo móvil	Sistema Operativo	Cliente de voz IP	Codecs
<b>100</b>	Smartphone Blackberry 9900	7.0 paquete 2435 v7.0.0.598 plataforma 5.0.0.743	Join Express Blackberry Edition V.1.6.0.1	PCM ley $\mu$ PCM ley A AMR
<b>101</b>	Smartphone Blackberry 9900	7.0 paquete 1672 v7.0.0.353 plataforma 5.0.0.544	Join Express Blackberry Edition V.1.6.0.1	PCM ley $\mu$ PCM ley A AMR
<b>102</b>	Samsung Galaxy SIII mini (GT- I8190L)	Android 4.1.2	Linphone Version 2.0.3	speex 16 khz speex 8 khz ilbc amr gsm g722 silk 16 khz pcmu pcma
<b>103</b>	Iphone 5	iOS 6.1.4	Linphone iPhone 2.0.4	speex 16 khz speex 8 khz silk 24 khz silk 16 khz ilbc gsm g722 pcmu pcma

Tabla 4.1 Dispositivos móviles para la realización de pruebas.

A continuación se presenta la captura de paquetes entre diferentes usuarios de dispositivos móviles.

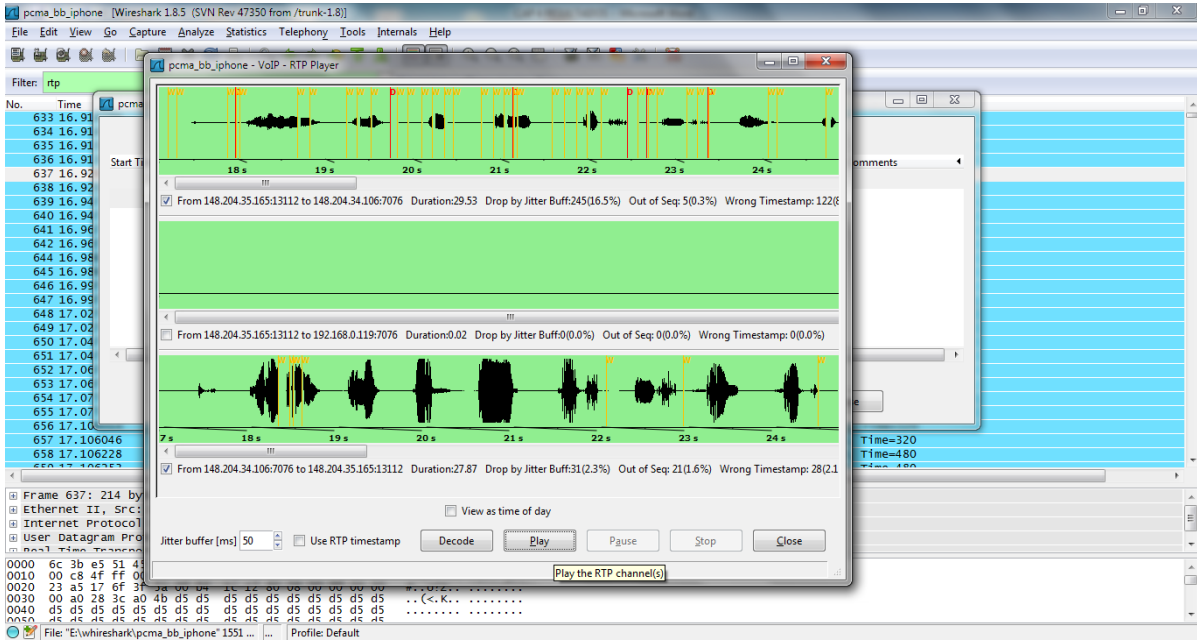


Figura 4.12 Llamada entre Blackberry e iPhone codec PCM ley A

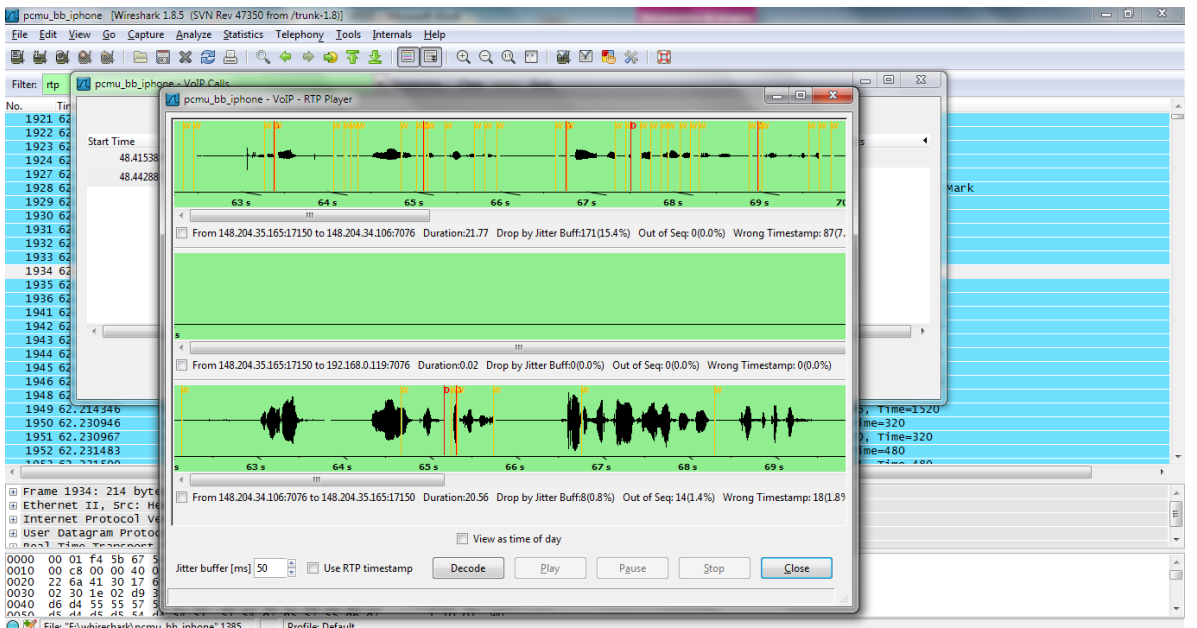


Figura 4.13 Llamada entre Blackberry e iPhone codec PCM ley μ



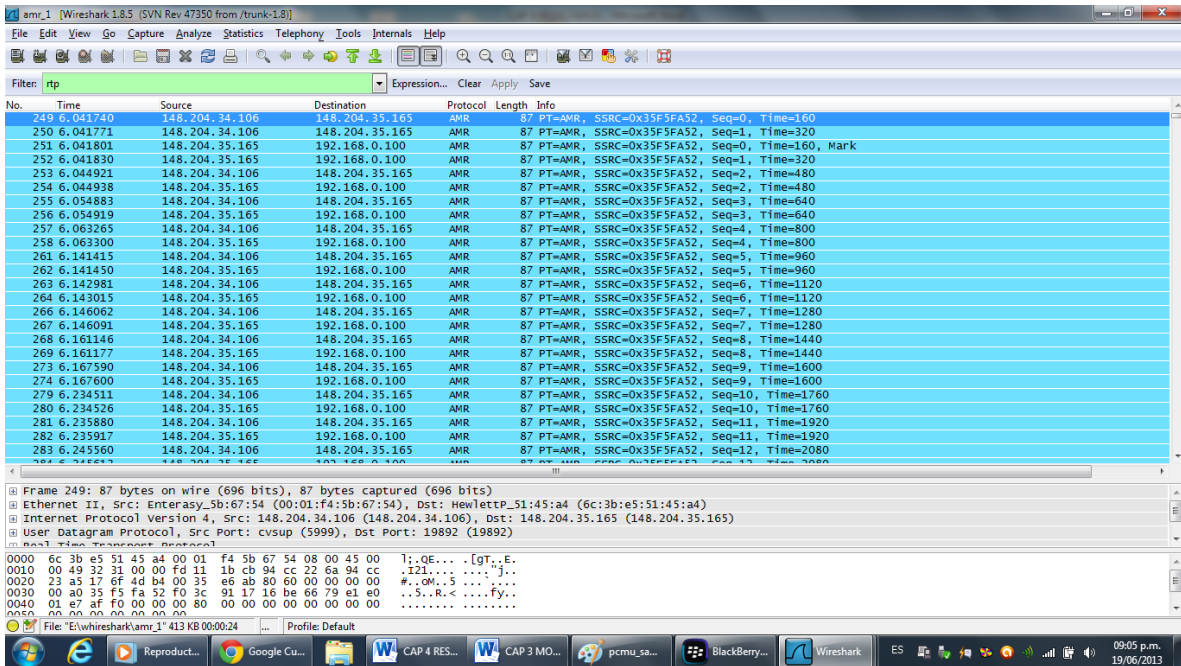


Figura 4.14 Llamada entre Blackberry codec AMR.

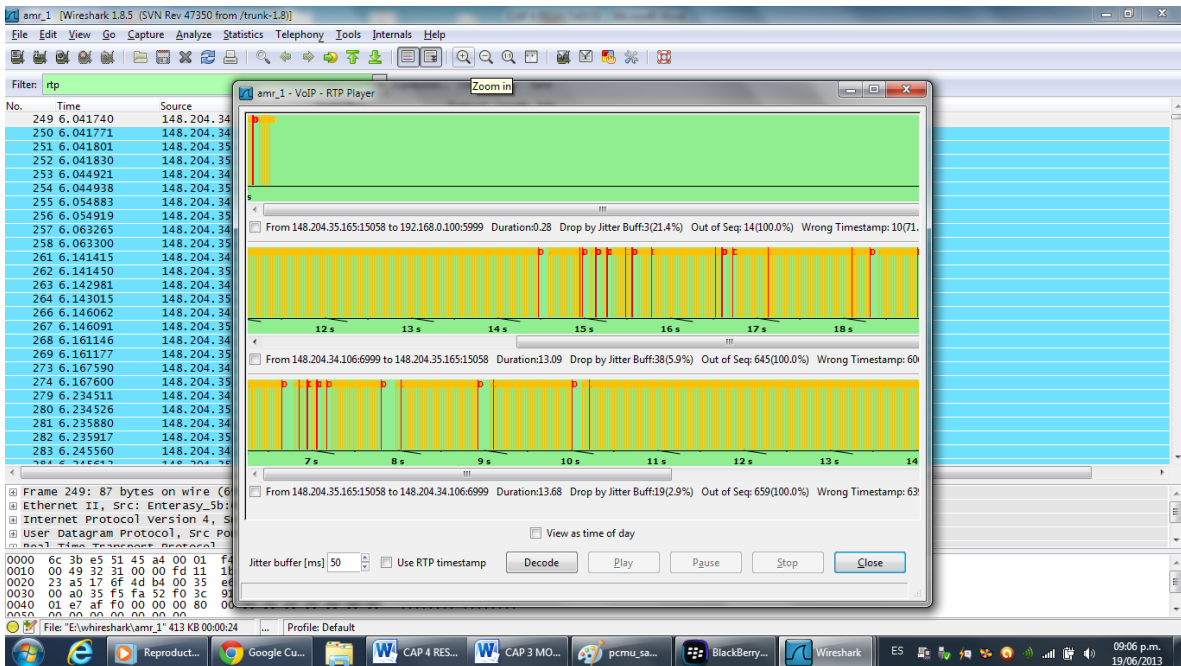


Figura 4.15 Decodificación de llamada entre Blackberry codec AMR.

En la figura 4.14 se muestra la captura de paquetes RTP entre usuarios con el codec AMR (*Adaptative Multi-Rate*), en la figura 4.15 se muestra la decodificación de los paquetes de voz, como podremos observar el región de color amarillo no se visualizan señales

senoidales que puedan ser interpretadas como señales de voz, esto se debe a que el codec AMR es un codec que requiere licencia y la aplicación whire shark entre los codecs preinstalados no cuenta con este codec, sin embargo al establecer entre usuarios mediante el codec AMR se puede establecer una llamada de excelente calidad y sin pérdida de paquetes.

Dispositivos		Smartphone Blackberry 9900			Samsung Galaxy SIII mini			Iphone 5	
		AMR	PCM ley $\mu$	PCM ley A	AMR	PCM ley $\mu$	PCM ley A	PCM ley $\mu$	PCM ley A
Smartphone Blackberry 9900	AMR	OK			OK				
	PCM ley $\mu$		OK			OK		OK	
	PCM ley A			OK			OK		OK
Samsung Galaxy SIII mini	AMR	OK			OK				
	PCM ley $\mu$		OK			OK		OK	
	PCM ley A			OK			OK		OK
Iphone 5	PCM ley $\mu$		OK			OK		OK	
	PCM ley A			OK			OK		OK

Tabla 4.2 Resumen de pruebas entre dispositivos móviles

En la tabla 4.2 se enlista un resumen de pruebas entre los diferentes dispositivos móviles, como podemos observar cuando realizamos llamadas entre dispositivos diferentes el único



requisito que debemos considerar es que estén configurados para el emplear el mismo codec, de otra forma no obtendremos comunicación entre estos dispositivos.

## Capítulo 5

# CONCLUSIONES Y TRABAJOS FUTUROS.

### 5.1 Conclusiones.

Como ya se explicó en el desarrollo de este trabajo existe una tendencia generalizada para la sustitución de los sistemas tradicionales de comunicación de voz, por sistemas de voz IP con sus respectivas ventajas y desventajas.

Por otro lado las comunicaciones móviles han tenido un auge impresionante, de tal forma que en estos momentos es más factible que un usuario cuente con un dispositivo móvil de comunicación, que uno fijo.

En este trabajo se comprobó la viabilidad para la implementación de un sistema de voz IP mediante un servidor de voz asterisk empleando dispositivos móviles con diferentes sistemas operativos, integrados estos mediante una red inalámbrica, con el fin de explotar las características móviles de esos dispositivos.

Dentro de la información recabada en el desarrollo del proyecto se observó que en la mayoría de los casos se emplea el codec G711 en sus dos versiones ( $\mu$  y A), en algunos otros el codec G729, considerando que este último es un codec propietario y por lo tanto es necesario adquirir el licenciamiento respectivo aunque asterisk cuenta con la versión G729a y puede ser empleada, esta no se encuentra actualizada nos puede proporcionar una referencia de su empleo y en caso de considerarse conveniente realizar la adquisición de las licencias que se requieran a razón del número de llamadas simultaneas empleando este codec.

Por otro lado y principalmente para los dispositivos de la marca BlackBerry se considera conveniente el empleo del codec AMR ya que al realizar llamadas empleando este codec se estableció una comunicación clara, comprensible y sin cortes, minimizando algunos errores característicos de la voz IP (*Perdida y retardo de paquetes*), en las aplicaciones Join express y linphone para Android se encuentra preinstalado este codec, no así en la aplicación linphone para iOS.

Debido a la gran demanda de dispositivos móviles de comunicación actualmente existe una demanda exponencial de aplicaciones para estos dispositivos, todas estas aplicaciones irán creciendo a la par de las investigación y desarrollo para voz IP por lo que se considera que en un futuro próximo exista un mayor número de aplicaciones de este tipo con sus respectivas mejoras, desde los entornos gráficos, hasta los codecs que se empleen para este tipo de sistemas.

## **5.2 Trabajos Futuros.**

Se considera conveniente la investigación y profundización en este campo de las comunicaciones, debido a que es muy amplio, incluye tenas de comunicaciones, redes de datos y programación orientada a objetos.

En este trabajo únicamente se empleó el servidor de comunicaciones como una central telefónica para el establecimiento de comunicaciones de voz, pero contiene muchas características que pueden ser explotadas como la contestadora automática, servidor de mensajes de texto, almacenamiento de mensajes de voz, servidor de correo electrónico y grabación de llamadas, por lo que se recomienda explotar estas características y de esta manera aprovechar los recursos informáticos adquiridos para la puesta en marcha del servidor de comunicaciones de voz.

Dentro de las características de la central telefónica existe el empleo de esquemas de seguridad para mantener la secrecía de las conversaciones mediante protocolos seguros en tiempo real (*SRTP y ZRTP*), en el caso de este trabajo no se emplearon estos protocolos debido a que los clientes de voz IP en estudios aún no se encuentran preparados para su empleo, los clientes de voz IP que cuentan con estos protocolos de seguridad en la mayoría

de los casos tienen un costo por descarga inicial, así como un costo por renovación anual de licencias, haciendo inviable el estudio en este campo.

Para el caso de una entidad gubernamental o empresa que requiera este tipo de esquemas de seguridad existe la opción de adquirir clientes de voz IP compatibles con los protocolos SRTP y ZRTP o en su caso diseñar aplicaciones propias con esas características.

Por otro lado es conveniente realizar una serie de pruebas con las redes de datos que proporcionan los prestadores de servicio de telefonía celular y de esta manera realizar un análisis de desempeño para la implementación de sistemas de comunicación de voz mediante protocolos IP empleando la infraestructura instalada por estos proveedores de servicio, así como el estudio de desempeño de los codecs disponibles en las aplicaciones de voz IP.