

# Universidad Autónoma Metropolitana Unidad Azcapotzalco

División de Ciencias Básicas e Ingeniería  
Licenciatura en Ingeniería en Computación

Proyecto Tecnológico:

**Sistema de telefonía VoIP sobre una LAN**

**Trimestre 2014-P**

Proyecto que presenta:

**Jose Alberto Martinez Vazquez**

para obtener el título de:

**Ingeniero en Computación**

Asesor:

**M. en C. Oscar Alvarado Nava**

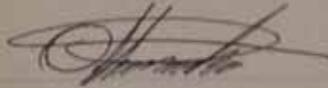
Ciudad de México

Agosto de 2014

Yo, **Oscar Alvarado Nava**, declaro que aprobé el contenido del presente Reporte de Proyecto de Integración y doy mi autorización para su publicación en la biblioteca Digital, así como en el Repositorio Institucional de UAM Azcapotzalco.



Yo, **Jose Alberto Martinez Vazquez**, doy mi autorización a la Coordinación de Servicios de Información de la Universidad Autónoma Metropolitana, Unidad Azcapotzalco, para publicar el presente documento en la Biblioteca Digital, así como en el Repositorio Institucional de UAM Azcapotzalco.



# Resumen

---

Se implementó un sistema de telefonía IP con software libre, utilizando Linphone (*soft-phones*) sobre computadoras convencionales con bocinas y micrófono.

El servidor PBX (*Private Branch Exchange*) se implementó con Asterisk, el cual se configuró para poder realizar llamadas entre usuarios de la misma y diferente LANs y ofrecer servicios como llamada en espera, correo de voz, transferencia de llamada, conferencias, entre otros.

Fue posible configurar una red telefónica entre LANs y WANs.



# Agradecimientos

---

- A la División de Ciencias Básicas e Ingeniería de la Universidad Autónoma Metropolitana, Unidad Azcapotzalco.
- Al Departamentos de Electrónica y de Sistemas
- A todos mis profesores de la Universidad.
- A mi madre Ma. Del Carmen Vazquez Sánchez.
- A mi asesor Oscar Alvarado Nava.
- A mis Amigos.
- Al amor de mi vida Ethel Ollivier Nuñez.



# Dedicatoria

---

*A mi asesor Oscar Alvarado por la ayuda que me ha dado durante este proceso y por los consejos de amigo que me ha dado.*

*A mi padre José Martínez Malvaez, gracias por tu apoyo y sé que estarás orgulloso de este logro donde quiera que estés. Descansa en paz.*

*A mis amigas y amigos Noemí Marín, Monserrat Alba, Luz Montoya, Janette Melendez, Sergio Ibáñez, Amaury Sandoval, etc., por pasar a mi lado los momentos de mi vida universitaria y estar conmigo siempre en las buenas y en las malas, jamás los olvidare*

*A mi Madre Carmen Vázquez Sánchez, por ser la amiga y compañera que me ayudo a crecer, gracias por estar siempre conmigo en todo momento. Gracias por la paciencia que has tenido para enseñarme, por el amor que me das, por tus cuidados en el tiempo que hemos vivido juntos, por los regaños que me merecía y que no entendía. Gracias mamá por estar durante toda esta etapa.*

*A la persona más especial que ha estado en mi vida; Ethel Ollivier Nuñez, gracias por todo tu amor y cariño que me demostraste, gracias por ser como eres, gracias por ser la mujer con mejores sentimientos que he conocido, gracias por ayudarme en todo el tiempo que estuviste conmigo; que nunca dejaste que me rindiera, gracias por hacerme crecer como persona, no sé cómo agradecerte y regresarte todo lo que me diste; y más ahora que estás tan lejos de mi. Siempre te llevaré en mi corazón. Te Amo mi Angelito.*

*Un agradecimiento especial para todas aquellas personas que de una y otra forma contribuyeron a este triunfo.*



# Índice general

---

Resumen	III
Agradecimientos	v
Dedicatoria	vii
Lista de Figuras	x
Lista de Tablas	xi
<b>1. Introducción</b>	<b>1</b>
1.1. Módulo de red . . . . .	2
1.2. Módulo PBX . . . . .	2
1.3. Módulo Softphone . . . . .	3
1.4. Módulo de Características . . . . .	3
1.5. Objetivos . . . . .	5
1.6. Antecedentes . . . . .	6
1.7. Justificación . . . . .	6
1.8. Organización del proyecto . . . . .	7
<b>2. VoIP</b>	<b>9</b>
2.1. Conmutación de circuitos . . . . .	10
2.2. Conmutación de paquetes . . . . .	10
2.3. Características de VoIP . . . . .	11
2.4. Elementos de una llamada . . . . .	14
2.5. Funcionalidad . . . . .	14
<b>3. SIP</b>	<b>15</b>
3.1. Funciones SIP . . . . .	16
3.2. Beneficios del protocolo SIP frente otros protocolos . . . . .	17
3.3. Arquitectura SIP . . . . .	18
<b>4. PBX</b>	<b>23</b>
4.1. PBX y PABX . . . . .	24
4.2. Funcionamiento . . . . .	24

4.3. Asterisk . . . . .	26
4.3.1. Asterisk como PBX . . . . .	27
4.3.2. Plataformas Soportadas . . . . .	28
4.3.3. Hardware Soportado . . . . .	28
4.3.4. Protocolos Soportads . . . . .	28
<b>5. Softphone</b>	<b>29</b>
5.1. Linphone . . . . .	31
<b>6. Instalación y Configuración</b>	<b>33</b>
6.1. Instalación de Asterisk . . . . .	33
6.2. Configuración de Asterisk . . . . .	39
6.2.1. Alta de Usuarios . . . . .	39
6.2.2. Correo de Voz . . . . .	40
6.2.3. Transferencia de Llamadas . . . . .	41
6.2.4. Captura de llamadas . . . . .	41
6.2.5. Video Llamadas . . . . .	42
6.2.6. DND (Estado de No Molestar) . . . . .	43
6.2.7. Música en Espera . . . . .	44
6.2.8. Salas de Conferencia . . . . .	45
6.2.9. Gestión de Llamadas . . . . .	46
6.2.10. CallBack . . . . .	47
6.2.11. Informe de Llamadas (CDR) . . . . .	48
6.3. Instalación de Linphone . . . . .	50
6.3.1. Windows . . . . .	50
6.3.2. Linux . . . . .	51
6.4. Configuración de Linphone . . . . .	51
6.5. Configuración de la red . . . . .	54
6.5.1. Switch . . . . .	54
6.5.2. Router . . . . .	55
<b>7. Resultados y conclusiones</b>	<b>59</b>
7.1. Resultados . . . . .	59
7.2. Conclusiones . . . . .	60
<b>A. Archivos de Configuración de Asterisk</b>	<b>69</b>

# Índice de figuras

---

1.1.	Diagrama de interacción de módulos. . . . .	1
1.2.	Diagrama de una red LAN. . . . .	2
3.1.	Establecimiento de la Comunicación Mediante SIP. . . . .	20
3.2.	Arquitectura del Protocolo SIP Para Establecer la Comunicación. . .	21
6.1.	Código de Área Correspondiente a México. . . . .	34
6.2.	Menu para añadir modulos de mysql. . . . .	37
6.3.	Permiso de Instalacion de Windows. . . . .	51
6.4.	Pantalla de Inicio de Linphone. . . . .	52
6.5.	Manage SIP Accounts. . . . .	52
6.6.	Configure SIP Account. . . . .	53
6.7.	Contraseña del Servidor. . . . .	53
6.8.	Usuario Reconocido por Asterisk. . . . .	54
7.1.	Transcurso de una llamada en linphone. . . . .	59
7.2.	Llamada entrante en linphone. . . . .	60
7.3.	Proceso de llamada en la consola de Asterisk. . . . .	61
7.4.	Protocolo SIP en Wireshark. . . . .	62
7.5.	Protocolo RTP en Wireshark. . . . .	63
7.6.	Protocolo UDP en Wireshark. . . . .	64
7.7.	Protocolo UDP en Wireshark. . . . .	65



# Índice de tablas

---

4.1. Ventajas de usar Asterisk. . . . .	27
5.1. Tipos de Softphone. . . . .	30
5.2. Características de Softphones. . . . .	31



# Capítulo 1

## Introducción

---

El proyecto fue compuesto de 4 módulos, que corresponde a cada uno de los objetivos del proyecto. El módulo principal fué el servidor PBX ya que es el encargado de gestionar al módulo de características y también será el encargado de realizar la comunicación (o conmutación) entre los Softphone. En la Figura 1.1 se muestra la relación entre módulos.

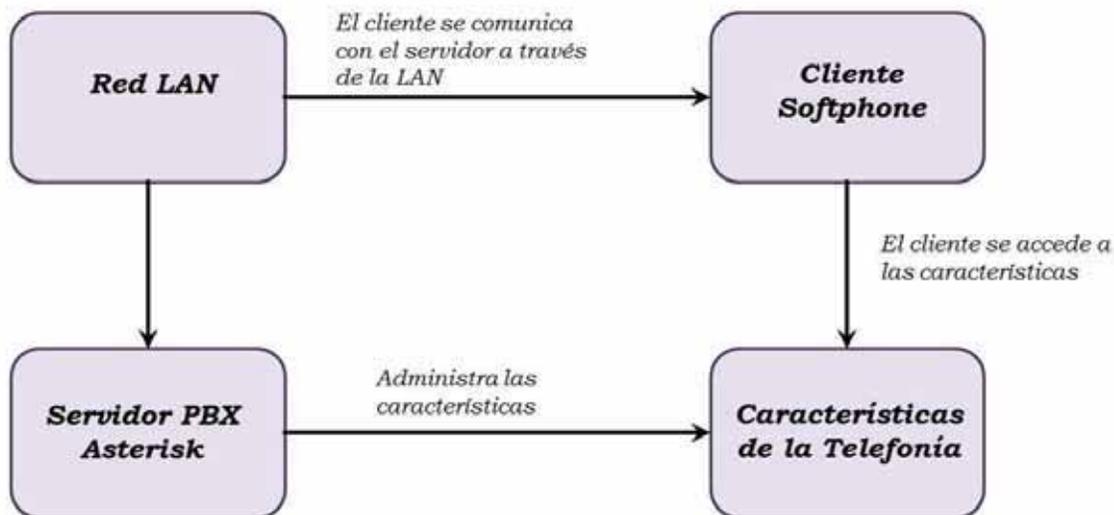


Figura 1.1: Diagrama de interacción de módulos.

La funcionalidad de cada uno de los módulos serán descritos a continuación.

## 1.1. Módulo de red

Este módulo se encarga de la comunicación de los equipos involucrados en el sistema; que son los usuarios y el servidor. También se hará una simulación de una red en donde los usuarios están en diferentes lugares geográficos (En distintas ciudades, localidades, edificios, etc.) como se muestra en la Figura 1.2.

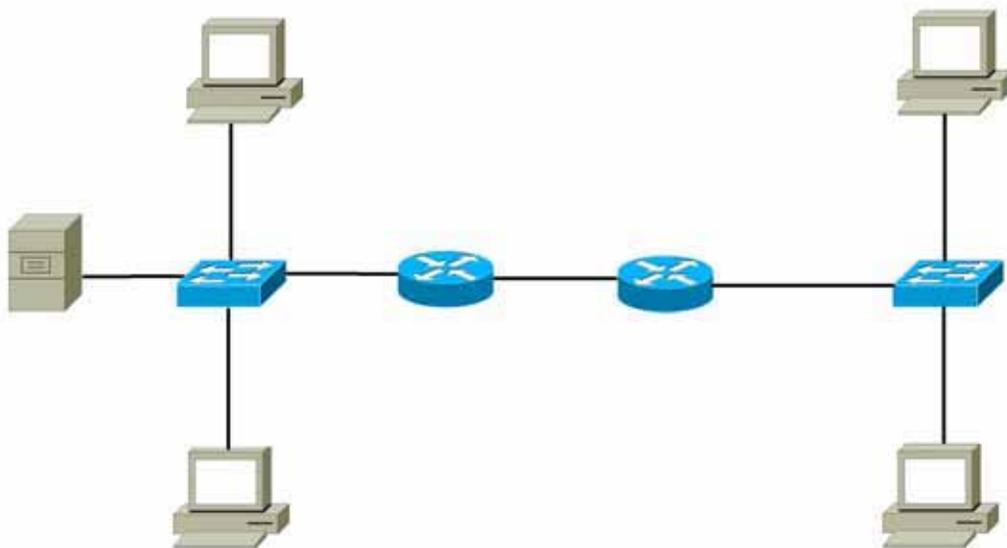


Figura 1.2: Diagrama de una red LAN.

## 1.2. Módulo PBX

Un PBX (*Private Branch Exchange*) se refiere al dispositivo que actúa como una ramificación de la red telefónica comutada pública (RCTP), por lo que los usuarios no se comunican directamente al exterior mediante líneas telefónicas convencionales, sino que al estar el PBX directamente conectado a la RTCP, será esta misma la que enrute la llamada hasta su destino final mediante enlaces unificados de transporte de voz llamados líneas troncales. En otras palabras, los usuarios de un PBX no están asociados con la central de teléfonos pública, ya que es la misma PBX la que actúa como tal, análogo a una central pública que de cobertura a todo un sector mientras que un PBX lo ofrece generalmente en las instalaciones de una compañía.

Los servidores PBX manejan un amplio rango de operaciones complejas de base de datos, tanto en tiempo real como fuera de él. Estas operaciones incluyen validación de usuarios, contabilidad, tarifas, recolección, distribución de utilidades, enrutamiento, administración general del servicio, carga de clientes, control del servicio, registro de usuarios y servicios de directorio entre otros.

## 1.3. Módulo Softphone

Un softphone es una aplicación de software para realizar llamadas telefónicas a través de una red de datos mediante un computadora de propósito general, en lugar de utilizar un hardware dedicado. A menudo, un teléfono virtual está diseñado para comportarse como un teléfono tradicional, a veces aparece como una imagen de un teléfono, con un panel de visualización y botones con los que puede interactuar el usuario. Un softphone se utiliza generalmente con un auricular conectado a la tarjeta de sonido del PC o un teléfono USB<sup>1</sup>.

Normalmente, un softphone es una parte de un entorno de Voz sobre IP y puede estar basado en el estándar SIP/H.323. o ser de origen privado. Existen varias implementaciones disponibles.

En este módulo se hace la simulación de un teléfono convencional a nivel de software, ya que fué instalado en las computadoras que tendrá cada usuario que este dentro de la red y quiera utilizar el servicio.

## 1.4. Módulo de Características

Este módulo será implementado dentro del servidor y será el encargado de gestionar y administrar las siguientes características.

**Transferencia de llamadas:** Éste servicio permite que se pueda transferir una llamada que tenga en la línea hacia otro número o extensión, y funciona de la siguiente manera:

1. Avisar a la persona con quien está hablando que va a realizar la transferencia de la llamada.
2. Oprimir y soltar rápidamente el gancho del teléfono. (Esto es, colgar y descolgar rápidamente su teléfono).
3. Envío del tono de activación de función.

---

<sup>1</sup>USB, el Universal Serial Bus (bus universal en serie) es un estándar industrial que define los cables, conectores y protocolos usados en un bus para conectar, comunicar y proveer de alimentación eléctrica entre ordenadores y periféricos y dispositivos electrónicos.

4. Marcar el número al cual desea transferir la llamada.
5. Esperar a que conteste la persona a la que le desea transferir la llamada. En cuanto lo haga avisarle sobre la transferencia de la llamada.
6. Colgar.

**Captura de llamadas:** La captura permite contestar una llamada que se está recibiendo en una extensión desde otra distinta. Normalmente esta opción se marca un código y después el número de la extensión desde donde se desea capturar la llamada.

**Conferencia múltiple:** En función del modelo de terminal se podrá establecer una comunicación entre múltiples usuarios de la centralita.

**DND:** Con esta característica el usuario podrá poner el teléfono en modo de no molestar, es decir, cuando entre una llamada el teléfono no sonara.

**Correo de Voz:** Es un sistema centralizado de manejo de mensajes telefónicos para un gran grupo de personas. Permite a los usuarios recibir, almacenar y gestionar mensajes de voz de las personas que le llaman cuando se encuentra ausente o con la línea ocupada.

Los buzones de voz se han convertido en una funcionalidad siempre presente en los sistemas telefónicos de casi todas las compañías.

**Música en espera:** Permite introducir categorías de música en espera basadas en archivos wav y mp3. De este modo podremos poner diferentes melodías para ser reproducidas como música en espera.

**Sala de conferencias:** Una sala de conferencias es un sistema que permite conectar a múltiples usuarios en una misma conversación telefónica. Los usuarios pueden acceder a la sala desde una extensión interna, o bien desde el exterior (a través de un número directo o bien a través de un IVR). Es un sistema muy útil para hacer reuniones internas (por ejemplo seguimiento de ventas). No hay una capacidad máxima de las salas por lo que podemos definir tantas como nos hagan falta (la limitación principal serán los recursos del servidor).

**Gestión de llamadas:** Esta característica es para definir un horario y calendario laboral que nos permita hacer un tratamiento diferenciado de las llamadas entrantes según el mismo. Por ejemplo, si estamos en horario laboral y no es festivo, la llamada se envía a la extensión 100 (de operadora). En caso contrario, se puede enviar a una locución que advierta de que son horas no laborables, o bien que es un día festivo.

**Callback:** Llamada automática de respuesta a una llamada perdida. Cuando redirigimos una llamada al módulo de callback el sistema lo que hará será colgar y originar

una llamada hacia el número que nos ha llamado, de este modo se pueden centralizar costes de llamada. La llamada saldrá siguiendo las normas de routing saliente de llamadas.

**Retro llamada:** Si se hace una llamada a una extensión y esta no contesta (por estar ocupado o ausente), se puede activar la función retro llamada. En cuanto el usuario llamado cuelgue, se avisa al que activo la función de retro llamada para que este pueda llamar de nuevo a la extensión inicial.

**Informes de llamadas:** Detalle de llamadas realizadas/recibidas por extensión, para imputación de costes departamentales, por cliente o incluso para facturación.

**Vídeo llamadas:** Permite la comunicación entre dos personas utilizando recursos de audio y vídeo dentro de los softphone, simultáneamente y en tiempo real.

## 1.5. Objetivos

El objetivo general de este proyecto es implementar un sistema de telefonía VoIP sobre una LAN, utilizando computadoras para realizar las llamadas dentro de la red o a números externos bajo el protocolo SIP.

A continuación se listan los objetivos específicos que se desarrollaron en este proyecto:

1. Diseñar e implementar una red LAN para realizar la conexión de los equipos.
2. Implementar un servidor PBX para la conexión de extensiones telefónicas dentro de la red.
3. Implementar Softphone para la simulación de teléfonos en una PC.
4. Implementar un conjunto de características de funcionalidad y/o personalización de los Softphone.

## 1.6. Antecedentes

Básicamente la VoIP, es un grupo de recursos que hacen posible que la señal de voz viaje a través de Internet empleando un protocolo IP (Protocolo de Internet). es un método en donde se toman señales de audio analógicas, que son las que se escuchan cuando uno habla por teléfono, a estas señales son transformadas en datos digitales que pueden ser transmitidos a través de internet hacia una dirección IP determinada.

En la actualidad la gran mayoría de las empresas cuentan con una red de computadoras (comunicación digital); y para realizar llamadas usan teléfonos convencionales (comunicación analógica), esto significa tener un cableado especial para las computadoras y otro para los teléfonos.

La telefonía VoIP es un sistema de telefonía que utiliza la red de computadoras para realizar la comunicación entre teléfonos sin la necesidad de tener un cableado único. Para ello se utilizara el protocolo de comunicación SIP para realizar la identificación de los paquetes utilizados en este sistema de telefonía.

## 1.7. Justificación

Hoy en día existen varios sistemas de telefonía VoIP, pero una gran parte de esta es de alguna compañía privada como los Cisco Call Manager<sup>2</sup> o teléfonos IP<sup>3</sup>, que generan costos para poder utilizar su tecnología.

En este proyecto se pretende realizar un sistema de telefonía con un servidor de software libre, y en vez de utilizar teléfonos IP se utilizaran computadoras convencionales que cuenten con bocinas y micrófono.

Los sistemas de telefonía tradicional están guiados por un sistema muy simple pero ineficiente denominado conmutación de circuitos. En este sistema cuando una llamada es realizada la conexión es mantenida durante todo el tiempo que dure la comunicación. Este tipo de comunicaciones es denominada "circuitos" porque la conexión esta realizada entre 2 puntos hacia ambas direcciones.

Esas llamadas son transmitidas a una calidad de 64kb por segundo (kbps) en cada dirección, por un total de transmisión de 128kb (64kb de ida y 64kb de vuelta). Como existen 8Kb en un KiloByte (KB), esto se transada en una transmisión de 16KB por cada segundo que el circuito esta abierto, y 960KB cada minuto que esta abierto. Entonces, en una comunicación de 10 minutos, el total transmitido seria de 9,600KB, lo que es prácticamente equivalente a 10 megas. Si observaras una conversación típica

---

<sup>2</sup>Es un software basado en un sistema de tratamiento de llamadas y telefonía sobre IP, desarrollado por Cisco Systems.

<sup>3</sup>Un terminal IP suele ser un dispositivo físico (similar a un teléfono normal).

te darías cuenta fácilmente que mucha de esta información es malgastada.

Cuando estas hablando, la otra parte esta escuchando, lo que significa que solo la mitad de la conexión se encuentra en uso en un momento dado. Basado en eso, podemos deducir que se podría cortar el tamaño de la conversación justo a la mitad, osea 4,7MB siempre manteniendo la misma calidad de comunicación.

La telefonía IP solo abre una pequeña conexión, suficientemente extensa para enviar una pequeña porción de información llamada paquete, de un sistema a otro:

- La computadora que envía divide la información en pequeños paquetes, con una dirección en cada uno indicando a los dispositivos de red donde enviar los mismos.
- Dentro de cada paquete hay una porción de la información que se esta enviando, en este caso la voz.
- La computadora emisora envía un paquete al router más cercano y se olvida del mismo. El router cercano envía el paquete a otro router que se encuentre mas cerca del destino, ese router se lo envía a otro que se encuentra todavía mas cerca del destino, ese a otro mas cerca, y así.
- La computadora receptora finalmente recibe los paquetes (que pueden haber tomado caminos completamente diferentes para haber llegado ahí). Usa las instrucciones contenidas en los paquetes para rearmar los datos en su estado original.

El intercambio de paquetes es muy eficiente. Deja a la red enviar los paquetes a lo largo de las rutas menos congestionadas. También libera a las computadoras de forma que estas pueden también aceptar información proveniente de otras computadoras.

## 1.8. Organización del proyecto

El reporte está organizado de la siguiente forma:

En el capítulo 2 se explica que es VoIP y los elementos necesarios que debe tener un sistema VoIP para que sea funcional.

En el capítulo 3 se explica que es Protocolo SIP y como es que funciona dentro de la VoIP, sus características además de las ventajas y desventajas con respecto a otros protocolos de voz.

En el capítulo 4 se explica que es un PBX, sus características y algunas de sus funciones que tiene este tipo de servidor. Además de cómo es la funcionalidad de un

sistema Asterisk.

En el capítulo 5 se explica que es un Softphone y las características que tienen, además de algunas de las ventajas que tiene frente a un teléfono IP. Además de algunos softphones que hay en este momento.

En el capítulo 6 se explica cómo es que hay que realizar la instalación y configuración de Asterisk, Linphone y la configuración de los equipos de red para que el sistema funcione correctamente.

En el capítulo 7 se explica y se muestran los resultados y conclusiones que se tuvieron al realizar este proyecto.

# Capítulo 2

## VoIP

---

VoIP proviene del inglés Voice Over Internet Protocol, que significa “voz sobre el protocolo de internet”. Básicamente VoIP es un método por el cual tomando señales de audio analógicas del tipo de las que se escuchan cuando uno habla por teléfono se las transforma en datos digitales que pueden ser transmitidos a través de internet hacia una dirección IP determinada.

El VoIP permite la unión de dos mundos históricamente separados, el de la transmisión de voz y el de la transmisión de datos. Entonces, el VoIP no es un servicio sino una tecnología. VoIP puede transformar una conexión estándar a internet en una plataforma para realizar llamadas gratuitas por internet. Usando algunos de los software gratuitos para llamadas VoIP que están disponibles en internet evitamos a las compañías tradicionales de telefonía, y por consiguiente, sus tarifas.

En el pasado, las conversaciones mediante VoIP solían ser de baja calidad, esto se vio superado por la tecnología actual y la proliferación de conexiones de banda ancha, hasta tal punto llegó la expansión de la telefonía IP que existe la posibilidad de que sin saberlo ya haya utilizado un servicio VoIP, por ejemplo, las operadoras de telefonía convencional, utilizan los servicios del VoIP para transmitir llamadas de larga distancia y de esta forma reducir costos.

Se sabe que va a llevar algún tiempo pero es seguro que en un futuro cercano desaparecerán por completo las líneas de teléfono convencionales que utilizamos en nuestra vida cotidiana, el avance tecnológico indica que estas serán muy probablemente reemplazadas por la telefonía IP.

## 2.1. Conmutación de circuitos

Los sistemas de telefonía tradicional están guiados por un sistema muy simple pero ineficiente denominado conmutación de circuitos. La conmutación de circuitos ha sido usada por las operadoras tradicionales por más de 100 años. En este sistema cuando una llamada es realizada la conexión es mantenida durante todo el tiempo que dure la comunicación. Este tipo de comunicaciones es denominado “c circuito” porque la conexión se lleva a cabo entre dos puntos hacia ambas direcciones. Estos son los fundamentos del sistema de telefonía convencional. Para entender cómo funciona una comunicación en telefonía IP primero vamos a definir cómo funciona una comunicación mediante el sistema de telefonía convencional de conmutación de circuitos.

1. Se levanta el teléfono y se escucha el tono de marcado. Esto deja saber que existe una conexión con el operador local de telefonía.
2. Se marca el número de teléfono al que se desea llamar.
3. La llamada es transmitida a través del conmutador (switch) del operador apuntando hacia el teléfono marcado.
4. Una conexión es creada entre el teléfono y la persona que se está llamando, en este proceso el operador de telefonía utiliza varios conmutadores para lograr la comunicación entre las 2 líneas.
5. El teléfono suena a la persona que estamos llamando y alguien contesta la llamada.
6. La conexión abre el circuito.
7. Uno habla por un tiempo determinado y luego cuelga el teléfono.
8. Cuando se cuelga el teléfono el circuito automáticamente es cerrado, de esta manera liberando la línea y todas las líneas que intervinieron en la comunicación.

## 2.2. Conmutación de paquetes

La conmutación de paquetes mantiene la conexión abierta y constante, el intercambio de paquetes que utilizan la telefonía IP solo abre una pequeña conexión, suficientemente extensa para enviar una pequeña porción de información llamada paquete, de un sistema a otro, esto funciona así:

- La computadora que emisora divide la información en pequeños paquetes, con una dirección en cada uno indicando a los dispositivos de red donde enviar los mismos.

- Dentro de cada paquete hay una porción de la información que se está enviando, la voz.
- La computadora emisora envía un paquete al router más cercano. El router cercano envía el paquete a otro router que se encuentre más cerca del destino, ese router se lo envía a otro que se encuentra todavía más cerca del destino, ese a otro más cerca, y así..
- Cuando la computadora receptora finalmente recibe los paquetes (que pueden haber tomado caminos completamente diferentes para haber llegado ahí). Usa las instrucciones contenidas en los paquetes para rearmar los datos en su estado original.
- El intercambio de paquetes es muy eficiente. Deja a la red enviar los paquetes a lo largo de las rutas menos congestionadas. También libera a las computadoras de forma que estas pueden también aceptar información proveniente de otras computadoras.

## 2.3. Características de VoIP

Utilizando VoIP no existe solo una sola forma de realizar una llamada, vamos a analizar las distintas opciones que nos presenta esta tecnología:

**ATA.** (Analog Telephone Adaptor) Esta es la forma más simple. Este adaptador permite conectar teléfonos comunes (de los que utilizamos en la telefonía convencional) a su computadora o a su red para utilizarlos con VoIP. El adaptador ATA es básicamente un transformador de analógico a digital. Este toma la señal de la línea de teléfono tradicional y la convierte en datos digitales listos para ser transmitidos a través de internet. Algunos proveedores de VoIP están regalando adaptadores ATA junto con sus servicios, estos adaptadores ya vienen pre configurados y basta con enchufarlos para que comiencen a funcionar.

**Teléfonos IP** (hardphones): Estos teléfonos a primera vista se ven como los teléfonos convencionales, con un tubo, una base y cables. Sin embargo los teléfonos IP en lugar de tener una ficha RJ-11 para conectar a las líneas de teléfono convencional estos vienen con una ficha RJ-45 para conectar directamente al router de la red y tienen todo el hardware y software necesario para manejar correctamente las llamadas VoIP. Próximamente, teléfonos celulares con Wi-Fi van a estar disponibles permitiendo llamadas VoIP a personas que utilicen este tipo de teléfonos siempre que exista conectividad a internet.

**Computadora a Computadora:** Esta es la manera más fácil de utilizar VoIP, todo lo que se necesita es un micrófono, parlantes y una tarjeta de sonido, además de una conexión a internet preferentemente de banda ancha. Exceptuando los costos del servicio de internet usualmente no existe cargo alguno por este tipo de comunicaciones VoIP entre computadora y computadora, no importa las distancias.

La primera ventaja y la más importante es el costo, una llamada mediante telefonía VoIP es en la mayoría de los casos mucho más barata que su equivalente en telefonía convencional.

Esto es básicamente debido a que se utiliza la misma red para la transmisión de datos y voz, la telefonía convencional tiene costos fijos que la telefonía IP no tiene, de ahí que esta es más barata. Usualmente para una llamada entre dos teléfonos IP la llamada es gratuita, cuando se realiza una llamada de un teléfono IP a un teléfono convencional el costo corre a cargo del teléfono IP.

Existen otras ventajas más allá del costo para elegir a la telefonía IP:

- Con VoIP uno puede realizar una llamada desde cualquier lado que exista conectividad a internet. Dado que los teléfonos IP transmiten su información a través de internet estos pueden ser administrados por su proveedor desde cualquier lugar donde exista una conexión. Esto es una ventaja para las personas que suelen viajar mucho, estas personas pueden llevar su teléfono consigo siempre teniendo acceso a su servicio de telefonía IP.
- La mayoría de los proveedores de VoIP entregan características por las cuales las operadoras de telefonía convencional cobran tarifas aparte. Un servicio de VoIP incluye:
  - Identificación de llamadas.
  - Servicio de llamadas en espera.
  - Servicio de transferencia de llamadas.
  - Repetir llamada.
  - Devolver llamada.
  - Llamada de 3 líneas (three-way calling).
- En base al servicio de identificación de llamadas existen también características avanzadas referentes a la manera en que las llamadas de un teléfono en particular son respondidas. Por ejemplo, con una misma llamada en Telefonía IP puedes:
  - Desviar la llamada a un teléfono particular.
  - Enviar la llamada directamente al correo de voz.
  - Dar a la llamada una señal de ocupado.
  - Mostrar un mensaje de fuera de servicio

Aun hoy en día existen problemas en la utilización de VoIP, queda claro que estos problemas son producto de limitaciones tecnológicas y se verán solucionadas en un corto plazo por la constante evolución de la tecnología, sin embargo algunas de estas todavía persisten y se enumeran a continuación.

- VoIP requiere de una conexión de banda ancha. Aun hoy en día, con la constante expansión que están sufriendo las conexiones de banda ancha todavía hay hogares que tienen conexiones por modem, este tipo de conectividad no es suficiente para mantener una conversación fluida con VoIP. Sin embargo, este problema se verá solucionado a la brevedad por el sostenido crecimiento de las conexiones de banda ancha.
- VoIP requiere de una conexión eléctrica. En caso de un corte eléctrico a diferencia de los teléfonos VoIP los teléfonos de la telefonía convencional siguen funcionando (excepto que se trate de teléfonos inalámbricos). Esto es así porque el cable telefónico es todo lo que un teléfono convencional necesita para funcionar.
- Llamadas al 911: Estas también son un problema con un sistema de telefonía VoIP. Como se sabe, la telefonía IP utiliza direcciones IP para identificar un número telefónico determinado, el problema es que no existe forma de asociar una dirección IP a un área geográfica, como cada ubicación geográfica tiene un número de emergencias en particular no es posible hacer una relación entre un número telefónico y su correspondiente sección en el 911. Para arreglar esto quizás en un futuro se podría incorporar información geográfica dentro de los paquetes de transmisión del VoIP. Dado que VoIP utiliza una conexión de red la calidad del servicio se ve afectado por la calidad de esta línea de datos, esto quiere decir que la calidad de una conexión VoIP se puede ver afectada por problemas como la alta latencia (tiempo de respuesta) o la pérdida de paquetes. Las conversaciones telefónicas se pueden ver distorsionadas o incluso cortadas por este tipo de problemas. Es indispensable para establecer conversaciones VoIP satisfactorias contar con una cierta estabilidad y calidad en la línea de datos.
- VoIP es susceptible a virus, gusanos y hacking, a pesar de que esto es muy raro y los desarrolladores de VoIP están trabajando en la encriptación para solucionar este tipo de problemas.
- En los casos en que se utilice un softphone la calidad de la comunicación VoIP se puede ver afectada por la PC, digamos que estamos realizando una llamada y en un determinado momento se abre un programa que utiliza el 100 % de la capacidad de nuestro CPU, en este caso crítico la calidad de la comunicación VoIP se puede ver comprometida porque el procesador se encuentra trabajando a tiempo completo, por eso, es recomendable utilizar un buen equipo junto con su configuración VoIP.

De cualquier forma, con la evolución tecnológica la telefonía IP va a superar estos problemas, y se estima que reemplace a la telefonía convencional en el corto plazo.

## 2.4. Elementos de una llamada

**Cliente:** El cliente establece y origina las llamadas voz, esta información se recibe a través del micrófono del usuario (entrada de información) se codifica, se empaqueta y, de la misma forma, esta información se decodifica y reproduce a través de los altavoces o audífonos (salida de la información).

Un Cliente puede ser un usuario de Skype o un usuario de alguna empresa que venda sus servicios de telefonía sobre IP a través de equipos como ATAs (Adaptadores de teléfonos analógicos) o teléfonos IP o Softphones que es un software que permite realizar llamadas a través de una computadora conectada a Internet.

**Servidores:** Los servidores se encargan de manejar operaciones de base de datos, realizado en un tiempo real como en uno fuera de él. Entre estas operaciones se tienen la contabilidad, la recolección, el enrutamiento, la administración y control del servicio, el registro de los usuarios. Usualmente en los servidores se instala software denominados Switches o IP-PBX (Conmutadores IP), ejemplos de switches pueden ser “VoIP-switch”, “Mera”, “Nextone” entre otros, un IP-PBX es Asterisk uno de los más usados y de código abierto.

## 2.5. Funcionalidad

VoIP puede facilitar tareas que serían más difíciles de realizar usando las redes telefónicas comunes:

- Las llamadas telefónicas locales pueden ser automáticamente enrutadas a un teléfono VoIP, sin importar dónde se esté conectado a la red. Uno podría llevar consigo un teléfono VoIP en un viaje, y en cualquier sitio conectado a Internet, se podría recibir llamadas.
- Números telefónicos gratuitos para usar con VoIP están disponibles en Estados Unidos de América, Reino Unido y otros países con organizaciones de usuarios VoIP.
- Los agentes de call center usando teléfonos VoIP pueden trabajar en cualquier lugar con conexión a Internet lo suficientemente rápida.
- Algunos paquetes de VoIP incluyen servicios extra por los que PSTN (Red Pública Telefónica Conmutada) normalmente cobra un cargo extra, o que no se encuentran disponibles en algunos países, como son las llamadas de 3 a la vez, retorno de llamada, remarcación automática, o identificación de llamada.

# Capítulo 3

## SIP

---

SIP (*Session Initiation Protocol*) es un protocolo de control y señalización usado mayoritariamente en los sistemas de telefonía IP, que fue desarrollado por el IETF (RFC 3261). Dicho protocolo permite crear, modificar y finalizar sesiones multimedia con uno o más participantes y sus mayores ventajas recaen en su simplicidad y consistencia.

La sintaxis de sus operaciones se asemeja a las de HTTP y SMTP, los protocolos utilizados en los servicios de páginas web y de distribución de e-mails respectivamente. Esta similitud es natural ya que SIP fue diseñado para que la telefonía se vuelva un servicio más en Internet.

Los clientes SIP usan el puerto 5060 en TCP (Transmission Control Protocol) y UDP (User Datagram Protocol) para conectar con los servidores SIP. SIP es usado simplemente para iniciar y terminar llamadas de voz y video. Todas las comunicaciones de voz/video van sobre RTP (Real-time Transport Protocol). Un objetivo de SIP fue aportar un conjunto de las funciones de procesamiento de llamadas y capacidades presentes en la red pública conmutada de telefonía. Así, implementó que permite en telefonía común como son: llamar a un número, provocar que un teléfono suene al ser llamado, escuchar la señal de tono o de ocupado. La implementación y terminología en SIP son diferentes.

Hasta la fecha, existían múltiples protocolos de señalización tales como H.323 de la ITU, el SCCP de Cisco, o el MGCP, pero parece que poco a poco SIP está ganando la batalla de estándar: Cisco está progresivamente adoptando SIP como protocolo en sus sistemas de telefonía IP en detrimento de H.323 y SCCP, Microsoft ha elegido SIP como protocolo para sus nuevos OCS (Office Communication Server), y los operadores (de móvil y fijo) también están implementando SIP dentro de su estrategia de convergencia, aprovechando de este modo la escalabilidad e interoperabilidad que nos proporciona el protocolo SIP.

SIP es un protocolo más nuevo que H.323 y no tiene madurez y soporte industrial. Sin embargo, por su simplicidad, escalabilidad, modularidad y comodidad con la cual integra con otras aplicaciones, este protocolo es atractivo para uso en arquitecturas de VoIP. Las sesiones pueden contener audio, video o datos. SIP solo maneja establecimiento, manejo y terminación de sesiones. Algunas de las características claves que SIP ofrece son:

- Descubrimiento dinámico de las capacidades media del endpoint, por uso del Protocolo de Descripción de Sesión (SDP).
- Descubrimiento dinámico de la disponibilidad del endpoint.
- Origen y administración de la sesión entre el host y los endpoints.

### 3.1. Funciones SIP

El protocolo SIP actúa de forma transparente, permitiendo el mapeo de nombre y la redirección de servicios ofreciendo así la implementación de la IN (Intelligent Network) de la PSTN o RTC.

Para conseguir los servicios de la IN el protocolo SIP dispone de distintas funciones. A continuación se enumeran las más importantes:

1. Localización de usuarios (SIP proporciona soporte para la movilidad).
2. Capacidades de usuario (SIP permite la negociación de parámetros).
3. Disponibilidad del usuario.
4. Establecimiento y mantenimiento de una sesión.

El protocolo SIP, permite establecer el procedimiento inicial de conexión para que dos UAs (User Agents) se conecten. UAs se llaman así a las terminales SIP que pueden ser teléfonos SIP, softphone gateways FXS/IP, Routers SIP, teléfonos USB, etc. Existen 2 tipos de UAs para la comunicación SIP los cuales son:

- User Agent Client (UAC): Origina las solicitudes SIP.
- User Agent Server (UAS): Responde a la solicitud de UAC.

Los UAC y UAS son capaces, sin los servidores de red de soportar una comunicación básica. Asumiendo el comportamiento de los componentes que a continuación se mencionan. Además de tener otros componentes involucrados en la comunicación como:

- Agente de Direcciones: Direccionan las solicitudes de las llamadas y retorna la dirección(es) de la parte llamada. En caso contrario rechaza la llamada, enviando una respuesta de error.

### 3.2. BENEFICIOS DEL PROTOCOLO SIP FRENTE OTROS PROTOCOLOS 17

- Servidor Proxy: Reenvía las solicitudes y respuestas para el establecimiento y liberación de las llamadas VoIP.
- Servidor de Registro: Registra las direcciones SIP y las direcciones IP asociadas.

El protocolo SIP adopta el modelo cliente-servidor y es transaccional. El cliente realiza peticiones (REQUEST) que el servidor atiende y genera una o más respuestas. Por ejemplo, para iniciar una sesión el cliente realiza una petición con el método (INVITE), en donde se indica que el usuario requiere establecer una conexión. El servidor responde, ya sea rechazando o aceptando la petición en una serie de respuestas, como se muestra en la Figura 3.1. Las respuestas llevan un código de estado, que brinda información acerca de las peticiones que fueron resueltas con éxito o si se produjo un error. La petición inicial y todas sus respuestas constituyen una transacción.

SIP como se ha visto, es un protocolo que provee mecanismos para la creación, modificación y finalización de sesiones. SIP funciona en combinación con SDP que es el encargado de la negociación de capacidades multimedia de los participantes involucrados, ancho de banda, negociación de los códecs, etc. Al ser SIP un protocolo solo de señalización, solo entiende del establecimiento, control y la terminación de las sesiones. En un protocolo simple, escalable y se integra con facilidad en otros protocolos. SIP puede funcionar sobre UDP y TCP, para VoIP se usará sobre UDP como se muestra en la Figura 3.2. Una vez establecida la sesión, los clientes intercambian directamente los contenidos multimedia de audio y/o video a través de, en este caso RTP. SIP tiene una estructura precida a HTML y SMTP. Esto lo vemos, por ejemplo, en que los clientes involucrados en una conexión tiene direcciones del tipo: usuario@dominio.

## 3.2. Beneficios del protocolo SIP frente otros protocolos

En la actualidad, los protocolos más usados en VoIP son tres:

- H.323 es un servicio estándar de la ITU que provee especificaciones para ordenadores, sistemas y servicios multimedia para redes que no proveen QoS (Calidad del Servicio). Como principales características de H.323 tenemos:
  - Implementa QoS de forma interna.
  - Control de conferencias.
- IAX2 ( Inter Asterisk eXchange) es un protocolo creado y estandarizado por Asterisk. Unas de sus principales características son la medida y señalización que viajen en el mismo flujo de datos.
  - Trunking.

- Cifrado de datos.

Una de las ventajas de este protocolo es que al enviar el "streaming" la señalización por el mismo flujo de datos, se evitan problemas derivados del NAT. Así pues, no es necesario abrir rangos de puertos para el tráfico RTP. Por último, IAX2 nos permite hacer trunking de forma que podemos enviar varias conversaciones por el mismo flujo, lo cual supone un importante ahorro de ancho de banda.

Finalmente, vemos qué hace de SIP un protocolo cada día más sólido. Aspectos importantes referentes a dicho protocolo se enumeran como sigue:

1. El control de llamadas es stateless o sin estado, y proporciona escalabilidad entre los dispositivos telefónicos y los servidores.
2. SIP necesita menos ciclos de CPU para generar mensajes de señalización de forma que un servidor podrá manejar más transacciones.
3. Una llamada SIP es independiente de la existencia de una conexión en la capa de transporte.
4. SIP soporta autenticación de llamante y llamado (emisor y receptor) mediante mecanismos HTTP.
5. Autenticación, criptografía y encriptación son soportados salto a salto por SSL/TLS pero SIP puede usar cualquier capa de transporte o cualquier mecanismo de seguridad de HTTP, como SSH o S-HTTP.
6. Un proxy SIP puede controlar la señalización de la llamada y puede bifurcar a cualquier número de dispositivos simultáneamente.

En definitiva, vemos que SIP es un protocolo con una gran escalabilidad, modular y muy apto para convertirse en el futuro inmediato de la VoIP.

### 3.3. Arquitectura SIP

El estándar define varios componentes SIP y hay varias formas de implementarlos en un sistema de control de llamadas.

- Servidor User Agent.
- Proxies.
- Registrars.
- Redirect.

- Location.

A menudo estos elementos son entidades lógicas que se ubican todas juntas para conseguir una mayor velocidad de procesamiento que dependerá a su vez de una buena configuración.

Normalmente los UAs son una aplicación en el ordenador del usuario, aunque a veces los UAs también pueden ser teléfonos móviles, PSTN gateways, una PDA, etc.

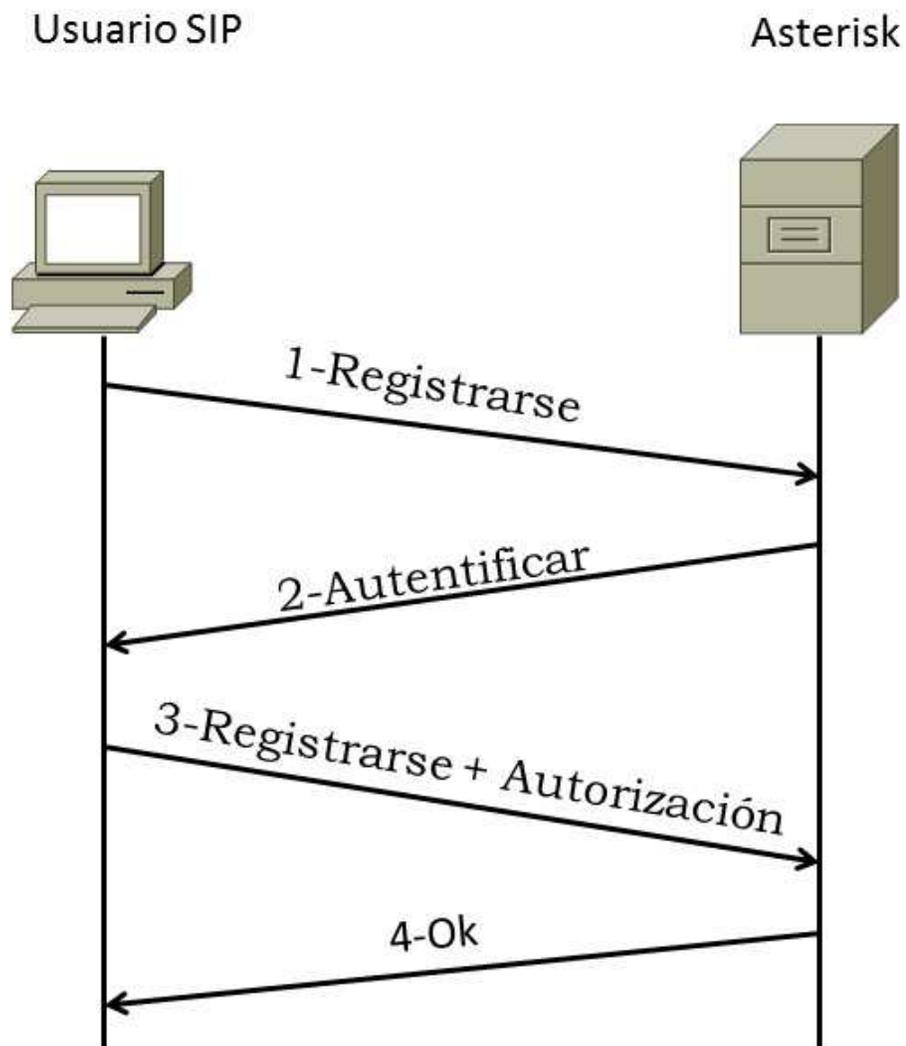


Figura 3.1: Establecimiento de la Comunicación Mediante SIP.

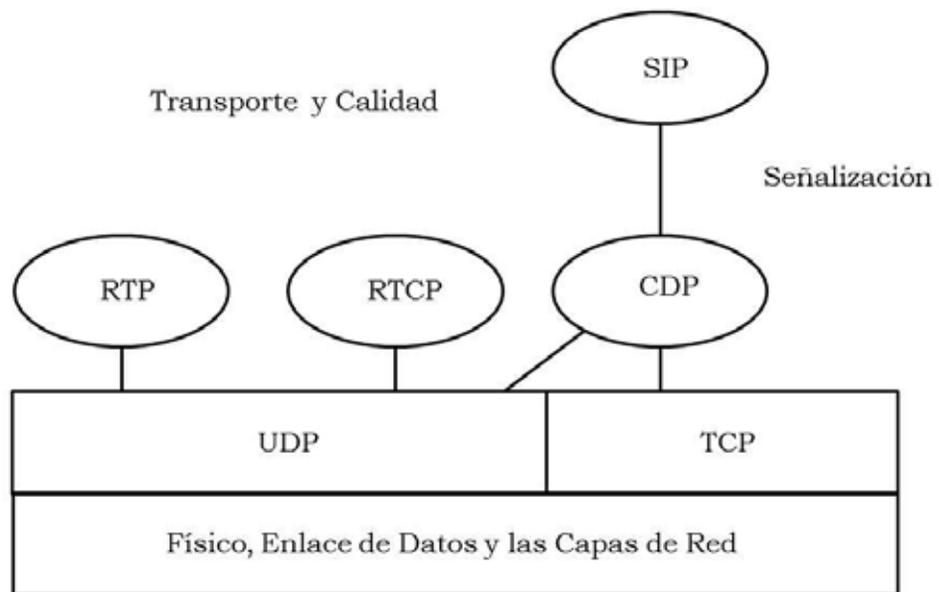


Figura 3.2: Arquitectura del Protocolo SIP Para Establecer la Comunicación.



# Capítulo 4

## PBX

---

Un PBX o PABX (siglas en inglés de Private Branch Exchange y Private Automatic Branch Exchange para PABX) cuya traducción al español sería Ramal Privado de Conmutación Automática, o más bien Central Secundaria Privada Automática; es en realidad cualquier central telefónica conectada directamente a la red pública de telefonía por medio de líneas troncales para gestionar además de las llamadas internas, las entrantes y salientes con autonomía sobre cualquier otra central telefónica. Este dispositivo generalmente pertenece a la empresa que lo tiene instalado y no a la compañía telefónica, de aquí el adjetivo Privado a su denominación.

Un PBX se refiere al dispositivo que actúa como una ramificación de la red primaria pública de teléfonos, por lo que los usuarios no se comunican directamente al exterior mediante líneas telefónicas convencionales, sino que al estar el PBX directamente conectado a la RTC (Red Telefónica Pública), será esta misma la que enrute la llamada hasta su destino final mediante enlaces unificados de transporte de voz llamados líneas troncales. En otras palabras, los usuarios de una PBX no están asociados con la central de telefonía pública, ya que es la misma PBX la que actúa como tal, análogo a una central pública que da cobertura a todo un sector mientras que un PBX lo ofrece generalmente en las instalaciones de una compañía.

Erróneamente se le llama PBX a cualquier central telefónica aunque no gestione las llamadas externas, bastando solo con que conmute líneas exteriores pertenecientes a otra central, que sí estaría conectada a la RTC. Estas serían centrales híbridas: Estas gestionan llamadas y enlazan líneas internas o extensiones pero al momento de comunicarse a un destino exterior, tan solo interconectaría la terminal con una línea convencional de la compañía de teléfono, mientras que un PBX se encargaría de procesar directamente el número marcado hacia el procesador central de la ciudad.

## 4.1. PBX y PABX

Del término PBX se deriva PABX, que no es más que un PBX automático.

En los orígenes de la telefonía era necesario conectar manualmente cables para establecer la comunicación. Este sistema era conocido como PMBX (PBX Manual) que luego fue reemplazado por un dispositivo electromecánico automático y posteriormente con el avance de la electrónica de microprocesadores, por sistemas digitales de conmutación que se le llamó PABX que desplazó al PMBX hasta casi inexistente. A partir de ese momento PABX y PBX se convirtieron en sinónimos.

los PBX (manuales) eran antiguas centrales telefónicas instaladas dentro del establecimiento comercial que la poseía. Requerían de un operador telefónico, o simplemente operador, para que realizase las funciones de conmutado de llamadas. A partir de ahora, cuando mencionemos a un PBX, se dará por hecho que es automático. No le denominaremos PABX debido a la mayor popularidad del primero.

Los PBX han pasado por una transformación rápida a través de los años, estas PBX fueron de analógicas, a semi-digitales y luego a totalmente digitales, ahora se están utilizando también las llamadas PBX IP, una PBX IP maneja las señales de voz bajo el protocolo de internet, aportando beneficios para la integración de telefonía informática (CTI). Un PBX IP puede existir como hardware físico, o puede llevar a cabo sus funciones de forma virtual, realizar las actividades e enrutamiento de llamadas de la PBX tradicional o PBX híbrido como un software. Esta versión virtual también se conoce como "SOFT PBX".

En muchos países de América Latina las compañías denominan como PBX a Centrex al servicio de asociar varias líneas de teléfono bajo un mismo número. Centrex no es más que el servicio simulado de una PBX por parte de la compañía de teléfono. Esto genera confusión porque son incluso las mismas compañías telefónicas quienes ofrecen el servicio bajo tales denominaciones. El término correcto a este servicio es "número telefónico único.º similares".

## 4.2. Funcionamiento

Los PBX de gran escala, instalados en grandes oficinas funcionan como dispositivos físicos que administran el tráfico de llamadas, incluso contabiliza las llamadas para uso financiero y de facturación (Información utilizada por la compañía de teléfono, quien programará la central además para hacerla compatible y reconozca otras centrales de su entorno de redes telefónicas primarias): Hoy en día un PBX no es más que una computadora especializada, siendo el usuario quien podrá configurar los parámetros de las llamadas entrantes y salientes. Generalmente el usuario conecta el PBX por un

único enlace digital, como E1 o T1, utilizando tan sólo 2 pares de cables en lugar de  $2n$  hilos para las  $n$  líneas externas como para degradarla más 2 líneas más que ocupan para el envío y recepción de información.

Las compañías locales generalmente imponen altos costos al servicio de enlace E1 o T1 y en algunos casos, aparte de cobrar altas tarifas fijas mensuales, factura cada minuto entrante o saliente; además requiere contratos especiales con la operadora telefónica. Sin embargo en países más desarrollados, no existen cargos por mantener un enlace E1 o T1, y por el contrario, se está migrando a estos enlaces en PBX que poseen infraestructura para tal en vez de usar líneas convencionales para desocupar cables en las acometidas intra-urbanas y descongestionar puertos en la central pública, ya que el PBX se vuelve en sí "su propia central".

El método de conexión para pequeñas y medianas empresas no deja de ser por líneas comunes de la compañía telefónica, utilizando cuantas líneas quiera tener el usuario, y éstas a su vez conectadas a la central telefónica, que a pesar de que podría tratarse de un PBX, no estaría funcionando como tal, y tan solo como una central privada híbrida. Esto se debe a que el tráfico de la llamada entrante o el inicio de la llamada saliente se originan en la central pública de la empresa de telefonía, probablemente al igual que otros abonados de la zona, mientras que si trabajaste con PBX, el tráfico de llamadas culminaría a se iniciaría en la misma centralita. Hay casos en que se usa un enlace digital con capacidad de varias llamadas simultáneas con la red exterior y aparte un par de líneas convencionales con fines ya sea de respaldo, o de estrategias de logística para dirigir llamadas por otras líneas de otros operadores que ofrecen tarifas más convenientes por cierto tipo de llamadas (Internacionales, teléfonos móviles, etc.). Por ejemplo; una empresa grande con muchos abonados tienen un enlace E1 con la empresa estatal o privada de teléfonos con capacidad de 30 líneas de entrada y 30 de salida utilizando un PBX con autonomía sobre la red troncal; 6 líneas conectadas a la red de telefonía móvil ya sea por bases (terminales fojos que simulan una línea convencional) o con dispositivos especializados que enlazan a la centralita con la red GSM con una tarjeta SIM con el fin de aprovechar tarifas preferentes llamando de móvil móvil; 2 líneas para llamadas internacionales conectadas como líneas convencionales ofrecidas por otra empresa con mejores tarifas.

Las nuevas tecnologías de telefonía IP (VoIP) permiten la conmutación de voz vía internet o redes informáticas privadas, siendo éste a veces el sistema de conexión del PBX con la red de telefonía local pública.

Las extensiones suelen ser líneas sencillas conectadas a teléfonos simples, con características similares a una línea de la RTC en cuanto a tensión y señales eléctricas, por lo que son perfectamente compatibles. Otros teléfonos pueden ser exclusivamente para ser usados con la marca del PBX, siendo compatibles sólo con estos. Estos últimos tienen funciones especiales adicionales como un display y se pueden monitorizar

todas las líneas o extensiones mediante LEDs de control; incluso se puede programar el PBX completamente desde ciertos modelos. Estos teléfonos requieren generalmente 4 hilos en sus conexiones mientras que las extensiones sencillas requieren sólo 2.

Se puede incluso conectar una extensión de un PBX a un puerto línea externa de otra PBX, por ejemplo, el de una oficina vecina para lograr comunicación interna gratuita. Pero lo más conveniente es usar interfaces ISDN para interconectar más de un PBX, pudiendo compartir llamadas internas y hasta líneas troncales, haciendo como si se tratara de una sola central, creando una mini-red telefónica. Los bancos y agencias con sucursales dispersadas en varios kilómetros comparten estas características y la voz viaja vía red local en paquetes de datos.

Existen varios estándares de interfaces con la RTC, con otros PBX o con otros teléfonos internos menos comunes.

### 4.3. Asterisk

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

Mark Spencer, de Digium, inicialmente creó Asterisk y actualmente es su principal desarrollador, junto con otros programadores que han contribuido a corregir errores y añadir novedades y funcionalidades. Originalmente desarrollado para el sistema operativo GNU/Linux, Asterisk actualmente también se distribuye en versiones para los sistemas operativos BSD, Mac OS X, Solaris y Microsoft Windows, aunque la plataforma nativa (GNU/Linux) es la que cuenta con mejor soporte de todas.

Asterisk incluye muchas características que anteriormente sólo estaban disponibles en costosos sistemas propietarios PBX, como buzón de voz, conferencias, IVR, distribución automática de llamadas, y otras muchas. Los usuarios pueden crear nuevas funcionalidades escribiendo un dialplan en el lenguaje de script de Asterisk o añadiendo módulos escritos en lenguaje C o en cualquier otro lenguaje de programación soportado en GNU/Linux.

Para conectar teléfonos estándares analógicos son necesarias tarjetas electrónicas telefónicas FXS o FXO fabricadas por Digium u otros proveedores, ya que para conectar el servidor a una línea externa no basta con un simple módem.

Quizá lo más interesante de Asterisk es que reconoce muchos protocolos VoIP como pueden ser SIP, H.323, IAX. Asterisk puede inter-operar con terminales IP ac-

tuando como un registrador y como gateway entre ambos.

Uno de los puntos fuertes del software Asterisk es que permite la unificación de tecnologías: VoIP, GSM y PSTN.

Asterisk se empieza a adoptar en algunos entornos corporativos como una gran solución de bajo coste junto con SER (SIP Express Router). Algunas de sus principales ventajas para usar Asterisk se muestran en la Tabla 4.1.

<b>Principales Ventajas</b>	
Funcionalidad	Asterisk dispone de todas las funcionalidades de las grandes centralitas propietarias (Cisco, Avaya, Alcatel, Siemens, etc.). Desde las más básicas (desvíos, capturas, transferencias, multi-conferencias, etc.) hasta las más avanzadas (Buzones de voz, IVR, CTI, ACD...).
Escalabilidad	El sistema puede dar servicio desde 10 usuarios en una sede de una pequeña empresa, hasta 10.000 de una multinacional repartidos en múltiples sedes.
Competitividad en coste	No solo por ser un sistema de código abierto (Open Source) sino gracias a su arquitectura hardware: utiliza plataforma servidor estándar (de propósito no específico) y tarjetas PCI para los interfaces de telefonía, que por la competencia del mercado se han ido abaratando progresivamente.
Interoperabilidad y Flexibilidad	Asterisk ha incorporado la mayoría de estándares de telefonía del mercado, tanto los tradicionales (TDM) con el soporte de puertos de interfaz analógicos (FXS y FXO) y RDSI (básicos y primarios), como los de telefonía IP (SIP, H.323, MGCP, SCCP/Skinny). Eso le permite conectarse a las redes públicas de telefonía tradicional e integrarse fácilmente con centralitas tradicionales (no IP) y otras centralitas IP.

Tabla 4.1: Ventajas de usar Asterisk.

### 4.3.1. Asterisk como PBX

Asterisk puede ser configurado como el núcleo de una central telefónica IP, permitiendo intercambio de llamadas, y permitiendo conexión de llamadas con el mundo

exterior a través de IP, permite realizar llamadas analógicas con el Servicio Telefónico Tradicional así como realizar llamadas digitales.

### **4.3.2. Plataformas Soportadas**

Asterisk funciona en una amplia variedad de sistemas operativos incluyendo Linux, Mac OS X, Solaris de Sun, entre otros. Ofrece todas las características de una PBX, incluyendo características avanzadas de gama alta de PBX privados. La arquitectura de Asterisk es compatible con Voz sobre IP en diversos protocolos y puede inter-operar con casi todos equipos de telefonía.

### **4.3.3. Hardware Soportado**

Asterisk no necesita de hardware adicional para Voz sobre IP. Para interacción con equipo telefónico digital y analógico, Asterisk es compatible con numerosos dispositivos de hardware.

### **4.3.4. Protocolos Soportados**

Asterisk soporta una amplia gama de protocolos para el manejo de voz sobre interfaces de telefonía tradicional; entre ellas H323, SIP, MGCP y SCCP.

# Capítulo 5

## Softphone

---

Un softphone (en inglés combinación de software y de telephone) es un software que hace una simulación de teléfono convencional por computadora y es utilizado para realizar llamadas a otros softphones o a otros teléfonos convencionales usando un VoIP (Voz sobre IP) o ToIP (Telefonía sobre IP). A menudo un Softphone es diseñado para comportarse y verse como un teléfono convencional, con un panel y botones con los cuales el usuario puede interactuar. Esto no es un detalle menor, ya que la idea es que el usuario se sienta cómodo y familiarizado con la aplicación, como lo está con su teléfono hogareño.

Estos teléfonos permiten utilizar toda la potencia y funcionalidad de un servidor IP PBX: transferencias, parqueo de llamadas, desvíos avanzadas, etc. Por sus características, estos teléfonos son económicos, e incluso gratuitos; así como funcionales y móviles, ya que se integran perfectamente en los equipos portátiles de los usuarios. Estos teléfonos utilizan la propia tarjeta de sonido y un micrófono corriente para realizar las comunicaciones.

Algunos softphones están implementados completamente en software, que se comunica con las PABX a través de la (LAN) Red de Área Local - TCP/IP para controlar y marcar a través del teléfono físico. Generalmente se hace a través de un entorno de centro de llamadas, para comunicarse desde un directorio de clientes o para recibir llamadas. En estos casos la información del cliente aparece en la pantalla de la computadora cuando el teléfono suena, dando a los agentes del centro de llamadas determinada información sobre quién está llamando y cómo recibirlo y dirigirse a esa persona.

El softphone transforma la computadora en un teléfono multimedia, con capacidad de voz, datos e imagen. Con ellos es posible hacer llamadas hacia teléfonos convencionales a través de internet, generalmente por un pequeño importe, y también realizar llamadas "PC-PC" gratuitamente, que es el tipo más popular de llamada

VOIP actualmente ah dos tipos de Softphones como se muestra en la Tabla 5.1.

<b>Propietarios</b>	Protocolo estándar: SIP, H323, etc.
	Protocolos propietarios abiertos.
	Protocolos propietarios abiertos.
<b>Libres</b>	Protocolos estándar.
	Protocolos propios abiertos.

Tabla 5.1: Tipos de Softphone.

Sus características principales son:

- Integración con el entorno (Escritorio).
  - Icono en systray, dock.
  - Aviso visual de llamadas entrantes.
- Integración con plataformas de acceso y validación de usuarios (LDAP).
- Importación/Exportación de datos: libretas de contactos en XML.
- Soporte de varias conversaciones simultáneamente y en algunos casos de varias líneas.

Existen softphones para instalar en computadoras convencionales y para instalar en Smartphones con diferentes sistemas operativos tales como Android, iPhone, BlackBerry, Windows Mobile, Symbian, etc.

## 5.1. Linphone

Linphone es una aplicación VoIP disponible en ordenadores con Linux, Windows, equipos de Apple con Mac OS X, Android, y teléfonos móviles iPhone. Se utiliza el SIP para la comunicación y está disponible bajo licencia GPL. Linphone utiliza GTK + GUI para Linux y se puede también utilizar como una aplicación en modo consola. Es compatible con la telefonía mediante el uso de un proveedor de servicios de telefonía por Internet (ITSP).

Las principales características de este programa son:

- Permite la comunicación gratuita de voz, vídeo y mensajería instantánea.
- Funciona por el protocolo SIP, lo que le hace compatible con cualquier servidor SIP, incluso ellos mismos disponen de un servidor SIP donde funcionar.
- Multiplataforma.
- Soporta IPv6.
- Soporta multi-llamada, llamada en espera y transferir llamada.

En la Tabla 5.2 se muestran algunos de los softphones mas usados.

Softphone	Tecnología	Licencia	S.O.	Soporte IPv6	Soporte Video
Linphone	SIP	OpenSource	Windows, Linux, MacOSX, BlackBerry, Iphone, Android	Si	Si
Zoiper	SIP,IAX	Versión gratuita y Comercial	Windows, Linux, MacOSX, Solaris Web, Windows Mobile	No	No
3CX	SIP	Versión gratuita y Comercial	Windows, Iphone, Android	No	No
XLite	SIP	Versión gratuita y Comercial	Windows, MacOSX	No	No
Blink	SIP	Versión gratuita y Comercial	Windows, MacOSX(Cocoa), Linux(Qt)	No	No

Tabla 5.2: Características de Softphones.



# Capítulo 6

## Instalación y Configuración

---

### 6.1. Instalación de Asterisk

Para configurar los informes de llamada antes de instalar asterisk hay que instalarlo esto vamos a instalar libmysqlclient-dev y mysql-server si la Base de Datos va a ir en el mismo servidor, al hacer la instalación hay que definir una contraseña de root para la base de datos.

```
root@asterisk:~/home#apt-get install libmysqlclient mysql-server
```

Una opción popular para la instalación de Asterisk es descargar el código fuente y compilarlo usted mismo. Si bien esto no es tan fácil como usar de gestión de paquetes o de usar una distribución Linux basada en Asterisk, no le permiten decidir cómo Asterisk se construye, y qué módulos están contruidos Asterisk, y se descarga de la siguiente paguna.

```
http://asterisk.org/
```

Después de descargar el archivo tar.gz hay descomprimir el archivo desde la terminal como superusuario

```
root@asterisk: /home#tar -zxvf asterisk-11.X.Y
```

Crearé una carpeta con el nombre asterisk-11.X.Y entramos a la carpeta

```
root@asterisk: /home#cd asterisk-11.X.Y/
```

En donde se encuentra el código fuente para compilar, pero antes de esto hay que verificar las dependencias, el mismo archivo tiene un verificador de estas dependen-

cias, solo hay que entrar a la siguiente carpeta

```
root@asterisk: /asterisk-11.X.Y#cd contrib/scripts/
```

Y ejecutar el siguiente comando, que tiene las opciones:

- test: solo muestra las bibliotecas que se instalaran.
- install: instala las dependencias de los paquetes. Dependiendo de la distribución de Linux y la versión de Asterisk esto debe de ser suficiente
- install-upacakged: instala las dependencias que no cuentan con paquetes.

```
root@asterisk: /asterisk-11.X.Y/contrib/scripts#./install_prereq install
```

En algunos casos pedirá que se defina el código de área, en este caso es 52; que es el código que le corresponde a México como se muestra en Figura 6.1.

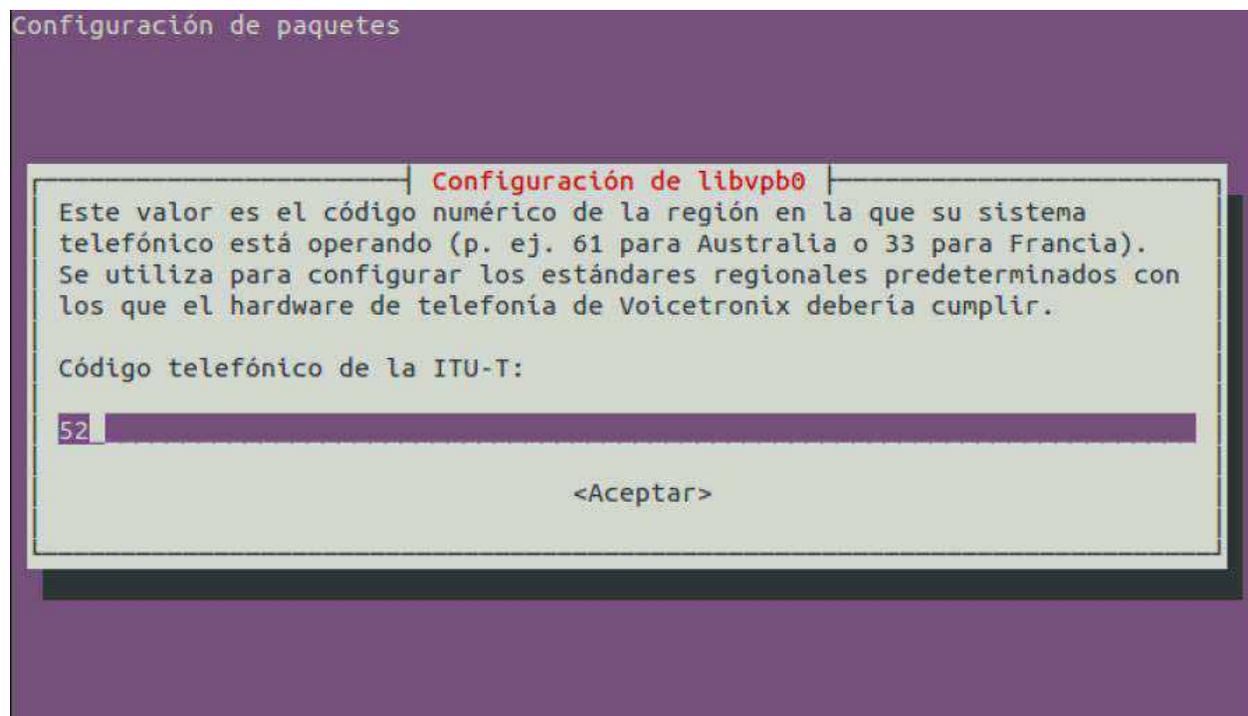


Figura 6.1: Código de Área Correspondiente a México.

Después de eso se regresa a la carpeta de Asterisk y se ejecuta el comando configure.

```
root@asterisk: /asterisk-11.X.Y#./configure
```

Al completar con éxito `./configure`, deberá ver el mensaje que es similar al que se muestra a continuación (Obviamente, el tipo de CPU host puede ser diferente al de abajo).

```

.          .$$$$$$$$$$$$$$$$$=..
.          .7$7..          .7$7:..
.          .7$7..          .7$7:..
.          .$$:.          ,7$.7
.          .7$.          7$$$$
.          ..$$$.          $$$$$
.          ..7$ .?.          $$$$$ .?.          7$$$$.
.          $.$.          .$$$7.          $$$$$7          .7$$$$.          .$$$..
.          .777.          .$$$$$$$77$$$$77$$$$$$$7.          $$$,
.          $$$~          .7$$$$$$$$$$$$$$$7.          .$$$..
.          .7$7          .7$$$$$$$7:          ?$$$..
.          $$$          ?7$$$$$$$$$$$I          .$$$7
.          $$$          .7$$$$$$$$$$$$$$$$$          :$$$..
.          $$$          $$$$$$7$$$$$$$$$$$$$          .$$$..
.          $$$          $$$          7$$$$7          .$$$          .$$$..
.          $$$$          $$$$7          .$$$..
.          7$$$$7          7$$$$          7$$$
.          $$$$          $$$
.          $$$7.          $$ (TM)
.          $$$$$$.          .7$$$$$          $$
.          $$$$$$$$$$$$$7$$$$$$$$$.          $$$$$
.          $$$$$$$$$$$$$$.

```

```

configure: Package configured for:&nbsp;
configure: OS type &nbsp;: linux-gnu
configure: Host CPU : x86_64
configure: build-cpu:vendor:os: x86_64 : unknown : linux-gnu :
configure: host-cpu:vendor:os: x86_64 : unknown : linux-gnu :

```

Ahora podemos compilar e instalar Asterisk. Para compilar Asterisk, sólo tiene que teclear `make` en la línea de comandos de Linux.

```
root@asterisk: /asterisk-11.X.Y#make
```

Para instalar los módulos de `mysql` hay que ejecutar el siguiente comando.

```
root@asterisk: /asterisk-11.X.Y#make menuselect
```

Aparecerá un menú en el cual hay que seleccionar el `Add-ons`, después hay que seleccionar los módulos `app-mysql` y `cdr_mysql` como se muestra en la Figura 6.2.

Ahora ha y que volver a ejecutar el comando `make` para que guarde los cambios. El paso de la compilación tardará varios minutos, y verá los distintos nombres de archivo desplazarse por lo que se están recopilando. Una vez que ha terminado de

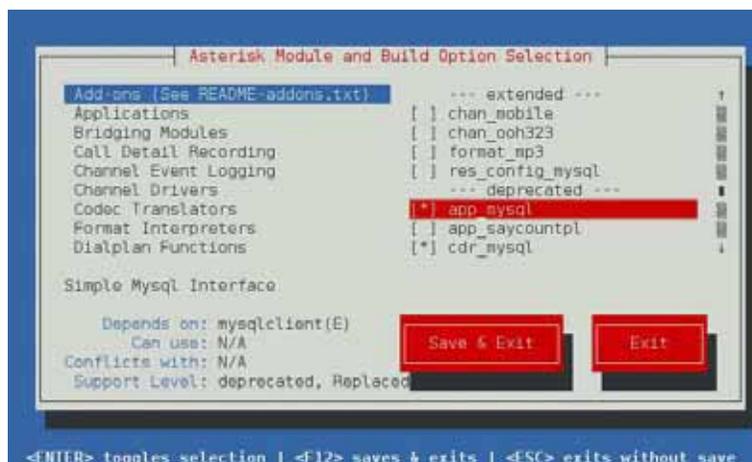


Figura 6.2: Menu para añadir módulos de mysql.

compilar Asterisk, verá un mensaje que se parece a este:

```
+----- Asterisk Build Complete -----+
+ Asterisk has successfully been built, and +
+ can be installed by running: +
+                                     +
+ make install +
```

A medida que el mensaje anterior indica, nuestro siguiente paso es instalar el programa y módulos de Asterisk compilada. Para ello, utilice el comando `make install`.

```
root@asterisk: /asterisk-11.X.Y#make install
```

Cuando haya terminado, Asterisk mostrará la siguiente advertencia:

```
+---- Asterisk Installation Complete -----+
+ +
+ YOU MUST READ THE SECURITY DOCUMENT +
+                                     +
+ Asterisk has successfully been installed. +
+ If you would like to install the sample +
+ configuration files (overwriting any +
+ existing config files), run: +
+ +
+ make samples +
+ +
+-----+
+---- Asterisk Installation Complete -----+
```

Instalación de archivos de ejemplo (no es un ejemplo de configuración de PBX). Para muchos de los archivos de configuración de ejemplo que hacen muestras de las instalaciones, la configuración contiene algo más que un ejemplo de configuración. Los archivos de configuración de ejemplo históricamente fueron utilizados predominantemente para la documentación de las opciones disponibles. Como tales, contienen muchos ejemplos de configuración de Asterisk que puede no ser ideal para las implementaciones estándar.

Durante la instalación de los archivos de configuración de ejemplo puede ser un buen punto de partida para algunas personas, no deben ser vistos como configuración recomendada para un sistema Asterisk.

Para instalar un conjunto de archivos de configuración de ejemplo para Asterisk, escribir el siguiente comando:

```
root@asterisk: /asterisk-11.X.Y#make samples
```

Los archivos de muestra existentes que han sido modificados se les darán una extensión de archivo. Por ejemplo, si usted tenía un archivo existente llamado `extensions.conf`, sería renombrado a `extensions.conf.old` y el plan de marcado de ejemplo se instala como `extensions.conf`.

Por defecto las grabaciones de asterisk son en inglés, para tenerlas en español hay que descargar el archivo `voipnovatoscoresoundsesgsm1.4.tar.gz` de la página:

```
http://www.voipnovatos.es/voces
```

Después de descomprimir este archivo hay que renombrar el directorio "es" a "en" dentro del directorio `voipnovatoscoresoundsesgsm1.4`, ahora desde la terminal hay que ingresar al directorio `/var/lib/asterisk/sounds`, ahora hay que renombrar el directorio "en" en "en.org".

```
root@asterisk: //var/lib/asterisk/sounds#mv en en.org
```

Ahora hay que mover las grabaciones en español al este directorio.

```
root@asterisk: //var/lib/asterisk/sounds#cp -r ../voipnovatoscoresoundses-  
gsm1.4/en /
```

## 6.2. Configuración de Asterisk

Para configurar los servicios de asterisk se modifican los archivos \*.conf que se encuentran en el directorio /etc/asterisk.

Antes de empezar a modificar estos archivos se recomienda hacer una copia de la carpeta anterior en una carpeta personal como Documentos y al editar los archivos borrar todo el contenido previo.

Para entrar a la consola de Asterisk se introduce el comando siguiente desde la terminal y para salir solo se introduce el comando exit.

```
root@asterisk: /etc/asterisk#asterisk -rvvvvvvv
```

Para revisar que cada función y modificación de los archivos funcione después de guardar cada archivo se tiene que reiniciar asterisk.

```
root@asterisk: /etc/asterisk#/etc/init.d/asterisk restart
```

En este manual solo se explica para configurar un solo usuario; se requiere realizar estas configuraciones para cada usuario que se agregue al sistema.

Nota: para poder realizar cambios en estos archivos se tiene que estar como root (superusuario).

Para que Asterisk pueda recibir solicitudes de usuarios en varias redes en el archivo sip.conf en el contexto general se añaden las siguientes líneas:

```
[general]                ;Contexto que aplica a todos los usuarios sip
bindaddr=0.0.0.0         ;Red donde se encuentra nuestro servidor
udpbindaddr=0.0.0.0     ;Escucha cualquier solicitud udp
tcpbindaddr=0.0.0.0     ;Escucha cualquier solicitud tcp
transport=udp           ;Tipo de paquetes que va a utilizar asterisk
```

### 6.2.1. Alta de Usuarios

El primer archivo a modificar es sip.conf para agregar los usuarios que estarán utilizando Asterisk, se crea un contexto para nuestras extensiones que estarán dentro del contexto general.

```
context=default        ;Contexto donde estarán los usuarios
```

```
[usuario1]           ;Nombre de usuario
type=friend         ;Para hacer y recibir llamadas
context=internal    ;Contexto donde se encuentra la extensión del usuario
host=dynamic        ;Tipo de host que tienen el usuario (IP)
secret=uam          ;Contraseña
allow=all           ;Tipo de codec de audio que se va a utilizar
```

Ahora para que nuestro usuario pueda utilizar su extensión editamos el archivo `extensions.conf`.

```
[internal]
;Contexto que utiliza el usuario
exten =>6001,1,Dial(SIP/usuario1,30)
;se define número de extensión, prioridad, protocolo utilizado, nombre de usuario, y
```

Para ver si nuestra extinción funciona se puede agregar lo siguiente en el mismo archivo, cuando el usuario marque a la extensión 1000 se reproducirá el archivo `hello-world` e inmediatamente colgara.

```
include =>aplicaciones
;esta linea se agrega en el contexto internal para que pueda utilizar otro contrtxto
[aplicaciones]
;contexto donde estaran nuestras funciones en asterisk
;Hola Mundo
exten =>1000,1,Answer()
;contesta la llamada entrante
exten =>1000,2,Wait(1)
;espera un segundo y pasa a la siguiente linea
exten =>1000,3,Playback(hello-world)
;reproduce la gravación
exten =>1000,4,Hangup()
;Curlga la llamada
```

### 6.2.2. Correo de Voz

Para la configuración del correo de voz primero se modifica el archivo `voicemail.conf`.

```
[default]
;contexto para el buzón de voz
7001=>7001,Alberto Martinez,usuariodominio.com
;asigna extensión de buzón =>contraseña(se recomienda que sea distinta
a la extensión), Nombre de usuario,correo donde se notificara que hay
un correo de voz
```

Ahora en nuestro archivo `sip.conf` se agrega la siguiente línea para cada usuario

que que vaya a tener el correo de voz y pueda tener acceso a su buzón.

```
mailbox=7001@default ;correo de voz buzón@contexto
```

Después se agrega el servicio en el archivo extensions.conf dentro del contexto aplicaciones.

```
;Correo de Voz
exten =>1100,1,Answer()
;contesta el correo de voz
exten =>1100,n,VoiceMailMain(@default)
;Después se ejecuta la aplicación del correo en donde se pedira usuario
y contraseña
```

### 6.2.3. Transferencia de Llamadas

Para realizar la transferencia de llamadas hay que editar el archivo features.conf y agregar las siguientes líneas.

```
[featuremap]
blindxfer =>## ;Transferencia sin atender
atxfer =>*2 ;Transferencia atendida
disconnect =>* ;Tecla de fin de llamada
automon =>*1 ;Teclas para grabar llamada
```

### 6.2.4. Captura de llamadas

Para realizar la captura de llamadas primero se tiene que modificar el archivo sip.conf y se agrega las siguientes líneas.

```
callgroup=1
;Definir un grupo de usuarios para separar áreas de la empresa
pickupgroup=1
;Solo se podrá capturar llamadas del grupo especificado, se puede
poner más de un grupo separado por ","
```

Después en el archivo features.conf se tiene que des comentar la siguiente línea o agregar según sea el caso, por defecto tiene \*8 si se desea se puede cambiar este valor por defecto.

```
Pickupexten = *8
;Teclas para capturar una llamada y después la extensión
```

Para verificar este cambio después de reiniciar el servicio, en la consola de Asterisk se debe introducir el comando.

```
CLI>features show
```

Y aparecerá algo similar a lo siguiente, esto quiere decir que ya está activa la captura de llamadas aquí también aparecerán las funciones de la transferencia de llamadas.

Builtin Feature	Default	Current
-----	-----	-----
Pickup	*8	*8
Blind Transfer	#	#
Attended Transfer		*2
One Touch Monitor		*1
Disconnect Call	*	*0
Park Call		*7
One Touch MixMonitor		*3

Después en el archivo `extensions.conf` se agregan las siguientes líneas dentro del contexto aplicaciones.

```
exten =>*8XXXX,1,Pickup(${EXTEN:2})
;Esto nos permitira capturar la llamada
same =>n,Hangup()
;Después de capturar la llamada se contesta
```

### 6.2.5. Video Llamadas

Para habilitar las video-llamadas; solo hay que añadir en la etiqueta general del archivo `sip.conf` la líneas.

```
[general]
context=default
videosupport=yes ;Para permitir video llamadas
;Codecs de video
allow=g729
allow=h261
allow=h263
allow=h263p
allow=h264
```

### 6.2.6. DND (Estado de No Molestar)

Abrimos el archivo extensions.conf y agregamos estas líneas.

```
[macro-DND]
exten =>s,1,GotoIf($DB_EXISTS(DND/${CALLERID(num)})?disable-dnd,1)
;verifica si hay un valor asociado al DND a nuestra extensión
exten =>s,n,Set(DB(DND/${CALLERID(num)})=1)
; si no existe el valor asociado se crea y asigna el valor de 1
exten =>s,n,playback(do-not-disturb)
; reproduce una grabación del DND
exten =>s,n,playback(activated)
; reproduce una grabación donde dice que está activado el DND
exten =>s,n,hangup
;cuelga la llamada
exten =>disable-dnd,1,Set(DEL=${DB_DELETE(DND/${CALLERID(num)})})
; si hay algún registro lo borrara
exten =>disable-dnd,n,playback(do-not-disturb)
; reproduce una grabación del DND
exten =>disable-dnd,n,playback(de-activated)
; reproduce una grabación donde dice que esta desactivado el DND
exten =>disable-dnd,n,hangup
; cuelga la llamada
```

En el mismo archivo dentro del contexto aplicaciones insertamos dos líneas más.

```
[aplicaciones]
exten =>99,1,Macro(DND) ;Para llamar al DND
exten =>99,n,Hangup ;Cuelga la llamada
```

La primera envía el llamante a la macro con nombre DND (la que acabamos de explicar más arriba). De esta forma podemos activar y desactivar el "No disturbar" llamando la extensión numero 99. Si el "No disturbar" está activado se desactivará de lo contrario se activará.

Ahora tenemos que decirle a Asterisk que para todas las llamadas entrantes haga este tipo de control.

Siempre en nuestro contexto aplicaciones, añadimos las siguientes líneas.

```
[local]
exten =>_2XXX,1,GotoIf($DB_EXISTS(DND/${EXTEN})?DND-ON)
exten =>_2XXX,n,Dial(SIP/${EXTEN},20,t)
exten =>_2XXX,n,GotoIf($['${DIALSTATUS}' = 'BUSY']?busy:unavail)
exten =>_2XXX,n(busy),Voicemail(${EXTEN}@default,b)
exten =>_2XXX,n,hangup
exten =>_2XXX,n(unavail),Voicemail(${EXTEN}@default,u)
exten =>_2XXX,n,hangup
exten =>_2XXX,n(DND-ON),Goto(2100,6)
exten =>h,1,Hangup
```

Si el registro existe en la base de datos de asterisk (línea 1) la llamada será enviada a la extensión con etiqueta DND-ON, y de ahí a la extensión \_2XXX prioridad 6. Esto significa que la llamada será transferida al contestador y al llamante será anunciado que la persona en la extensión XXXX no se encuentra disponible.

Si el registro no existe se procesará la línea dos y se llamará la extensión marcada. Luego según esté ocupada o en otra condición, la llamada se enviará a la correspondiente parte del plan de llamadas.

### 6.2.7. Música en Espera

Antes de empezar hay que crear la carpeta mohmp3 en el directorio /var/lib/asterisk, que es en donde guardaremos nuestra música.

```
root@asterisk: /var/lib/asterisk#mkdir mohmp3
```

Si existirá esta carpeta hay que copiarla y hacer un respaldo. Como la música en mp3 necesita muchos recursos del CPU hay que bajar la frecuencia y hacerla de audio mono con el comando sox.

```
root@asterisk: /#sox archivo.mp3 -c 1 -r 8000 archivo.mp3
```

Ahora copiamos nuestro archivo a la carpeta creada anteriormente.

```
root@asterisk: /#cp archivo.mp3 /var/lib/asterisk/mohmp3/
```

Ahora vamos a hacer que nuestra música pueda ser escuchada por los usuarios hay que editar el archivo musiconhold.conf.

```
[default]                ;Nuestro contexto
mode=files                ;Modo en que carga la música
directory=/var/lib/Asterisk/mohmp3 ;Directorio donde se encuentra
                             la música
random=yes                ;La música se reproduce
                             aleatoriamente
```

Para probar que funciona nuestra música hay que agregar la siguiente en el archivo `extensions.conf`.

```
exten =>1100,1,Answer()
same =>1100,n,MusicOnHold(default,30)
;El primer parametro indica el contexto de nuestra musica
El segundo indica el tiempo que reproducira la musica
exten =>2000,n,Hangup()
```

### 6.2.8. Salas de Conferencia

Para crear las salas de conferencias hay que editar el archivo `meetme.conf`, y hay que agregar las siguientes líneas dentro del contexto general para todas las salas.

```
[general]

audiobuffers=32
;Número de paquetes audio de 20ms que serán guardados en un buffer
de memoria cuando pertenecen a canales que no son DADHI. Esto permite
sincronizar el audio de los distintos participantes y evitar retrasos.
Debe ser un valor entre 2 y 32

schedule=yes
;con este parámetro le decimos a Asterisk que las conferencias pueden
ser programadas (solamente en realtime)

logmembercount=yes
; este parámetro sirve para actualizar el realtime cada vez que un
usuario entra/sale de una conferencia
```

```
fuzzystart=300
; Tenemos una conferencia programada a las 8 de la mañana. Este
parámetro sirve para definir si un usuario puede o no entrar a
la conferencia si esta no ha empezado. 300 es el número de
segundos. Ejemplo. Si el usuario entra a la 7:55 será aceptado
porque puede entrar hasta 5 minutos (300 segundos) antes del
comienzo de la conferencia. Si entra a las 7:50 será rechazado.
```

```
earlyalert=3600
; Este valor (siempre en segundos) define si al usuario que intenta
entrar en una conferencia programada le será anunciado o no que la
conferencia todavía no ha empezado. Si entra cuando falta menos de
una hora recibirá este aviso. Si entra más de una hora antes se le
anunciará que la conferencia no existe.
```

```
endalert=120
; cuando faltarán 120 segundo para terminar la conferencia programada,
le anunciará a todos los participantes.
```

Ahora dentro del mismo archivo se crean las salas que uno desee.

```
[rooms]
;Etiqueta donde se crean las salas de conferencias
conf =>8000
;Sala de conferencias abierta para todos los usuarios
conf =>8001,1234
;Sala de conferencias que pedirá un PIN para ingresar(1234)
conf =>8002,1234,5567
;Sala que también pedirá PIN de administrador
```

Finalmente configuramos el dialplan para la conferencia en el extensions.conf dentro de la etiqueta de aplicaciones.

```
exten =>1200,1,Goto(conf,1)
exten =>conf,1,Set(MEETME_RECORDINGFILE=/tmp/Tutorial-${TIMESTAMP})
exten =>conf,2,Meetme(8000,M)
exten =>conf,3,Hangup()
```

### 6.2.9. Gestión de Llamadas

Aquí configuraremos nuestro horario para realizar las llamadas, es decir cuando alguien llame fuera de horario laboral, para eso agregamos lo siguiente al archivo calendar.conf. Hay que definir un calendario para cada usuario.

```

[calendario]
;Etiqueta del calendario
type=caldav
;Tipo de calendario soportado: ical, caldav, exchange o ews
url=https://www.google.com/calendar/dav/usuario@gmail.com/events/
;URL de donde va a obtener el calendario
user=usuario@gmail.com
;usuario del servidor del calendario
secret=password
;contraseña para acceder al calendario
refresh=10
;actualiza el calendario cada n minutos
timeframe=180
;Extrae los eventos en los proximos N minutos
autoremind=10
;Anula cualquier notificacion y la cambia por los n minutos
channel=SIP/1000
;Envia una notificación del proximo evento
context=calendario
;nombre del contexto
extension=1010
;número de la extensión
waittime=45
;Tiempo para contestar la extensión

```

Por último se modifica el archivo extensions.conf.

```

[calendario]
exten =>1010,1,NoOp(Llamada desde el calendario)
same =>n,Agi(googletts.agi,"${CALENDAR_EVENT(summary)}.",es)
;contiene titulo del evento
same =>n,Agi(googletts.agi,.a las
${STRFTIME(${CALENDAR_EVENT(start)},,%H:%M)}.",es)
;contiene la hora del evento, la hora está en formato EPCH, hay
que transformarla en el formato de la función STRFTIME
same =>n,Wait(1)
same =>n,Playback(goodbye)
same =>n,Hangup

```

### 6.2.10. CallBack

El objetivo de lo que vamos a configurar es el siguiente, vamos a llamar a una extensión que estará ocupada, entonces marcaremos que queremos que nos llame cuando se quede libre, cuando finalmente la persona con la que queremos hablar cuelga el teléfono Asterisk nos generará una llamada, cuando descolgamos llamará a nuestro

interlocutor y nos pondrá en comunicación.

Bien lo primero que tendremos que modificar es el sip.conf y añadir en las extensiones que queramos que tengan este servicio lo siguiente:

```
cc_agent_policy=generic
;será el que se comunica con Asterisk y en nombre de quién Asterisk
actuará, la persona que indica que quiere que se rellame a alguien
cc_monitor_policy=generic
; la persona a la que se va a monitorizar por parte del llamante
para que en cuanto cuelgue le lancemos nuestra llamada.
```

Para poder realizar la petición de Callback tenemos la aplicación CallCompletionRequest que no tiene parámetros de entrada y que emite la petición a la última llamada que hemos realizado con estado Busy o No Response, si tras haber hecho la petición queremos cancelarla tenemos CallCompletionCancel.

```
exten =>99,1,CallCompletionRequest()
exten =>99,n,Hangup
exten =>98,1,CallCompletionCancel()
exten =>98,n,Hangup
```

Un PBX se refiere al dispositivo que actúa como una ramificación de la red primaria pública de teléfonos, por lo que los usuarios no se comunican directamente al exterior mediante líneas telefónicas convencionales, sino que al estar el PBX directamente conectado a la RTC (red telefónica pública), será esta misma la que enrute la llamada hasta su destino final mediante enlaces unificados de transporte de voz llamados líneas tronales. En otras palabras, los usuarios de una PBX no están asociados con la central de teléfonos pública, ya que es la misma PBX la que actúa como tal, análogo a una central pública que de cobertura a todo un sector mientras que un PBX lo ofrece generalmente en las instalaciones de una compañía.

### 6.2.11. Informe de Llamadas (CDR)

Para los informes de las llamadas se migrará a una base de datos en mysql (la actualización de los registros se hará de forma automática).

Ya instalado mysql hay que ingresar a mysql con el siguiente comando (pedirá la contraseña definida en la instalación).

```
root@asterisk: #mysql -u root -p
```

Dentro de mysql creamos la base de datos.

```
mysql>CREATE DATABASES asteriskCDR;
```

Creamos un usuario "asterisk" para esta database "asteriskCDR" e ingresamos a la base de datos creada.

```
mysql>GRANT INSERT ON asteriskCDR.* TO asterisk@localhost IDENTIFIED
BY 'password'
mysql>USE asteriskCDR;
```

Creamos la tabla dentro de la base de datos.

```
mysql>USE asteriskCDR;
mysql>CREATE TABLE cdr (
calldate datetime NOT NULL default '0000-00-00 00:00:00',
clid varchar(80) NOT NULL default '',
src varchar(80) NOT NULL default '',
dst varchar(80) NOT NULL default '',
dcontext varchar(80) NOT NULL default '',
channel varchar(80) NOT NULL default '',
dstchannel varchar(80) NOT NULL default '',
lastapp varchar(80) NOT NULL default '',
lastdata varchar(80) NOT NULL default '',
duration int(11) NOT NULL default '0',
billsec int(11) NOT NULL default '0',
disposition varchar(45) NOT NULL default '',
amaflags int(11) NOT NULL default '0',
accountcode varchar(20) NOT NULL default '',
userfield varchar(255) NOT NULL default '',
uniqueid VARCHAR(32) NOT NULL default '',
linkedid VARCHAR(32) NOT NULL default '',
sequence VARCHAR(32) NOT NULL default '',
peeraccount VARCHAR(32) NOT NULL default ''
);
```

Una vez hecho esto ponemos como índices para optimizar la búsqueda por estos campos.

```
ALTER TABLE cdr ADD INDEX ( 'calldate' );
ALTER TABLE cdr ADD INDEX ( 'dst' );
ALTER TABLE cdr ADD INDEX ( 'accountcode' );
```

Ahora hay que modificar el archivo cdr\_mysql.conf para que Asterisk pueda escribir sobre la base de datos, se ponen las siguientes líneas.

```
[global]
hostname=192.168.3.3
;IP de nuestra base de datos, si fuese local pondríamos localhost
dbname=asteriskCDR
;nombre de la base de datos
table=cdr
;nombre de la tabla
password=password
;contraseña definido en mysql para ingresar a la base
user=asterisk
; usuario para utilizar la base
port=3306
; puerto por donde se va a ingresar a la base
sock=/var/run/mysqld/mysqld.sock
;ruta del servicio de mysql

;se definen las columnas a ser llenadas en la base por asterisk
[columns]
;static =>
;alias =>
alias start =>calldate
alias callerid =>clid
;alias src =>src
;alias dst =>dst
;alias dcontext =>dcontext
;alias channel =>channel
;alias dstchannel =>dstchannel
;alias lastapp =>lastapp
;alias lastdata =>lastdata
;alias duration =>duration
;alias billsec =>billsec
;alias disposition =>disposition
;alias amaflags =>amaflags
;alias accountcode =>accountcode
;alias userfield =>userfield
;alias uniqueid =>uniqueid
```

## 6.3. Instalación de Linphone

### 6.3.1. Windows

Hay que ir a la pagina principal de linphone para descargar el archivo ejecutable linphone-3.7.0-setup.exe.

<http://www.linphone.org/eng/download/>

Después de haber descargado este archivo hay que instalarlo haciendo doble clic sobre él, y aparecerá la pantalla mostrada en la Figura 6.3.



Figura 6.3: Permiso de Instalacion de Windows.

Después de esto solo hay que seguir las instrucciones de instalación.

### 6.3.2. Linux

Linphone se instaló desde los repositorios de la distribución Debian, desde la terminal como superusuario, ya que es la manera más fácil de instalarlo en un sistema operativo Linux.

## 6.4. Configuración de Linphone

Ya instalado nuestro softphone en el cliente, hay que configurarlo para que Asterisk reconozca a nuestros usuarios, la pantalla de inicio de Linphone se muestra en la Figura 6.4.

En el menú de options hay que elegir Settings, y se abrirá una ventana. Después hay que seleccionar la pestaña Manage SIP Accounts como se muestra en la Figura 6.5.

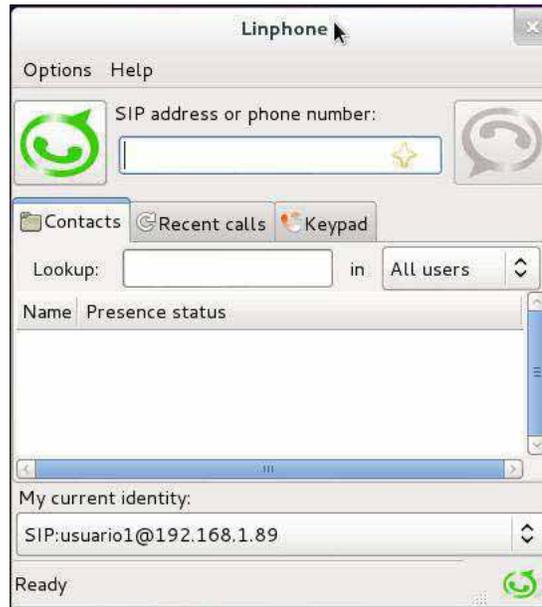


Figura 6.4: Pantalla de Inicio de Linphone.

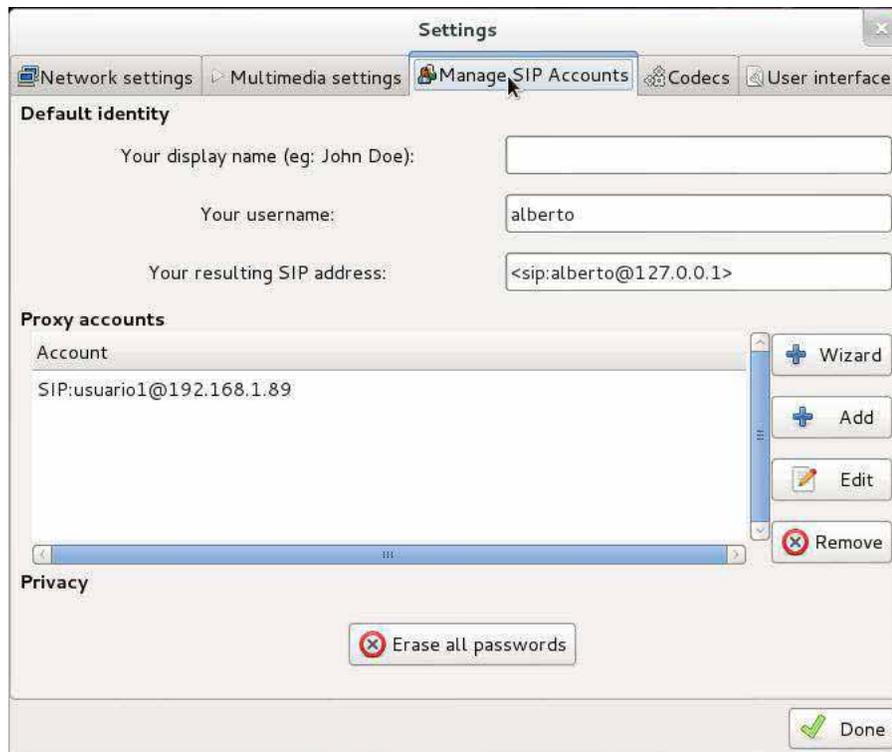


Figura 6.5: Manage SIP Accounts.

Después de esto hay que agregar a nuestro usuario en con el botón Add, abrirá una nueva ventana donde hay que definir nuestro usuario y la dirección IP de nuestro servidor como se muestra en Figura 6.6.

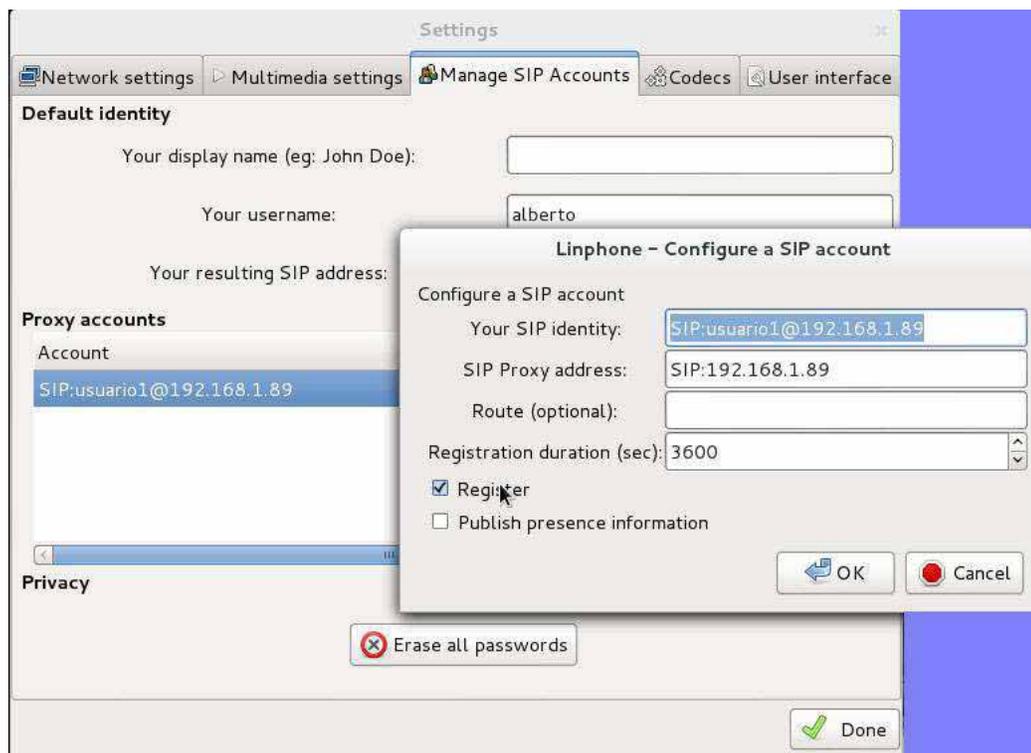


Figura 6.6: Configure SIP Account.

Ya solo hay que hacer clic en aceptar, después de esto nos pedirá la contraseña de usuario como se muestra en la Figura 6.7 definida en el archivo sip.conf cuando dimos de alta nuestro usuario.



Figura 6.7: Contraseña del Servidor.

Para verificar que nuestro Asterisk reconoce nuestro usuario dado de alta en el Softphone, en la parte inferior de éste aparecera nuestro usuario y junto a el un punto de color verde como se muestra en la Figura 6.8.

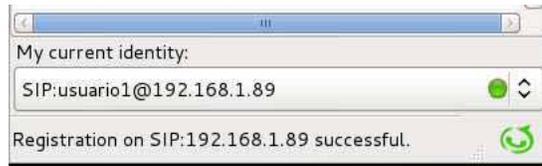


Figura 6.8: Usuario Reconocido por Asterisk.

## 6.5. Configuración de la red

### 6.5.1. Switch

Dentro del switch hay que entrar en modo privilegiado con el siguiente comando:

```
Switch>enable
```

Luego hay que entrar en el modo global para poder realizar las configuraciones con el comando:

```
Switch#configure terminal
```

Ahora se van a crear dos vlan para separar los paquetes de voz y los de datos, además de asignarle un nombre a cada vlan para poderlas identificar fácilmente con los siguientes comandos:

```
Switch(config)#vlan 2
Switch(config-vlan)#name Datos
Switch(config-vlan)#exit
Switch(config)#vlan 3
Switch(config-vlan)#name Voz
Switch(config-vlan)#exit
```

Después en cada interfaz en donde esté conectada una computadora o el servidor hay que asignarle nuestras vlans creadas, primero entramos a nuestra interfaz con el comando:

```
Switch(config)#interface fast0/10
```

Después lo ponemos en modo de acceso con el comando:

```
Switch(config-if)#switchport mode Access
```

Ahora le asignamos nuestra vlan 2 como la vlan de acceso con el comando:

```
Switch(config-if)#switchport access vlan 2
```

Después a la misma interfaz le asignamos nuestra vlan de voz que es la vlan 3 con el comando:

```
Switch(config-if)#switchport voice vlan 3
```

Ahora en la interfaz que va conectada al router, primero la seleccionamos con el comando:

```
Switch(config)#interface fastEthernet 0/1
```

Después le decimos que va a estar en modo troncal para que deje pasar las vlan que hemos creado con el comando:

```
Switch(config-if)#switchport mode trunk
```

Como en esta prueba solo se creó 1 vlan de datos, le decimos que nuestra vlan 2 va a ser la nativa con el comando:

```
Switch(config-if)#switchport native vlan 2
```

### 6.5.2. Router

Al igual que el switch hay que entrar en modo de configuración con los comandos:

```
Router>enable  
Router#configure terminal
```

Ahora ingresamos a la interfaz que va a estar conectado a nuestro switch con el comando:

```
Router(config)#interface FastEthernet0/0
```

Después le asignamos una IP con su máscara de red, (esta IP va a ser nuestra puerta de enlace de la red) con el comando:

```
Router(config-if)#ip address 192.168.1.1 255.255.255.0
```

A diferencia del switch, en el router hay que habilitar las interfaces con el comando:

```
Router(config-if)#no shutdown
Router(config-if)#exit
```

Ahora en nuestra interfaz serial que va conectada a otro router hay que seleccionar esta interfaz con el comando:

```
Router(config)#interface Serial0/0/0
```

Al igual que la interfaz anterior hay que asignarle una IP junto con su máscara (Nota: en el otro router que esté conectado a esta interfaz hay que asignarle una IP distinta que este dentro de la misma red, por lo general se recomienda que sea la siguiente IP disponible).

```
Router(config-if)#ip address 192.168.2.1 255.255.255.0
```

También a esta interfaz hay que habilitarla.

```
Router(config-if)#no shutdown
Router(config-if)#exit
```

Para estas pruebas se utilizó el ruteo estático ya que solo se conectaron 2 redes distintas, y le decimos a nuestro router que la red a la que se quiere comunicar junto con su máscara y la siguiente IP que ve nuestro router con el comando:

```
Router(config)#ip route 192.168.3.0 255.255.255.0 192.168.2.2
```

Con el siguiente comando se puede ver que nuestro ruteo estático es correcto.

```
Router#show ip route
```

Debe aparecer algo similar a la siguiente información, en la parte inferior muestra las redes a las que se tiene acceso desde nuestro router y cómo es que puede llegar a ellas.

```
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B -
BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
```

i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area  
\* - candidate default, U - per-user static route, o - ODR  
P - periodic downloaded static route  
Gateway of last resort is not set  
C  
192.168.1.0/24 is directly connected, FastEthernet0/0  
C  
192.168.2.0/24 is directly connected, Serial0/0/0  
S  
192.168.3.0/24 [1/0] via 192.168.2.2



# Capítulo 7

## Resultados y conclusiones

### 7.1. Resultados

Como se muestra en las figuras siguientes el funcionamiento del softphones cuando se está efectuando una llamada. En la Figura 7.1 se puede apreciar cuando otro usuario está llamando a nuestra extensión y en la Figura 7.2 se muestra cuando una llamada está en curso.

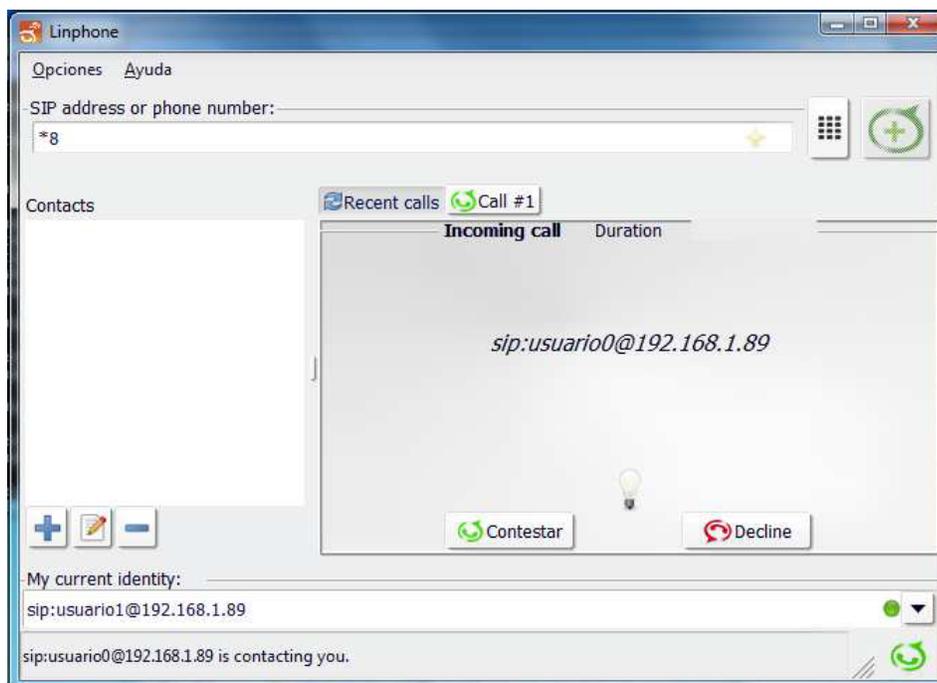


Figura 7.1: Transcurso de una llamada en linphone.



Figura 7.2: Llamada entrante en linphone.

Las llamadas también pueden ser monitoreadas desde la consola de Asterisk, en donde se puede apreciar el proceso de la llamada como se muestra en la Figura 7.3.

Además de que se realizó una captura de paquetes en Wireshark para ver la transferencia de los paquetes que hay entre los usuarios y el servidor; y los protocolos que está utilizando para él envío de estos paquetes son el SIP (Figura 7.4) para la señalización telefónica, RTP Figura 7.5 para la voz, UDP Figura 7.6 para la comunicación.

Para verificar que el informe de llamadas (CDR) funciona correctamente, se ingresa a mysql y al revisar el la tabla que se asignó, y ahí se ven los datos indicados como los usuarios involucrados en la llamada, la duración, la fecha y hora en que se realizó, entre otros como de muestra en la Figura 7.7.

## 7.2. Conclusiones

Como se menciona en la introducción y en los objetivos el servidor PBX fue instalado en Debian 7.5, los softphones fueron instalados sobre Debian 7.5 y Windows 7, aunque el softphone también está disponible para Mac OS y teléfonos móviles para los sistemas iOS, Windows Phone y Android.

La versión de Asterisk Instalada fue la 11.10 y la versión de Linphone utilizada fue la 3.7.0, ya que en momento en que fue instalado este software era la última versión estable. Asterisk ya instalado puede ser actualizado desde los repositorios.

```

Terminal (como superusuario)
Archivo Editar Ver Buscar Terminal Pestañas Ayuda
Terminal
-- Executing [6000@internal:1] Dial("SIP/usuario7-00000001", "SIP/usuario0,30") in new stack
== Using SIP RTP CoS mark 5
-- Called SIP/usuario0
-- SIP/usuario0-00000002 is ringing
-- SIP/usuario0-00000002 answered SIP/usuario7-00000001
-- Remotely bridging SIP/usuario7-00000001 and SIP/usuario0-00000002
> 0x9defcd8 -- Probation passed - setting RTP source address to 192.168.1.75:7076
> 0xalala78 -- Probation passed - setting RTP source address to 192.168.1.68:7078
> 0x9defcd8 -- Probation passed - setting RTP source address to 192.168.1.75:7076
-- Started music on hold, class 'default', on SIP/usuario7-00000001
> 0xalala78 -- Probation passed - setting RTP source address to 192.168.1.68:7078
-- Stopped music on hold on SIP/usuario7-00000001
-- Started music on hold, class 'default', on SIP/usuario7-00000001
[Aug 6 08:46:17] WARNING[3610]: chan_sip: 100 retrans_pk: Retransmission timeout reached on transmission sZopVDzlyN for seqno 103 (Critical Request) -- See https://wiki.asterisk.org/wiki/display/AST/SIP+Retransmissions
Packet timed out after 3200lms with no response
[Aug 6 08:46:17] WARNING[3610]: chan_sip: 100 retrans_pk: Hanging up call sZopVDzlyN - no reply to our critical packet (see https://wiki.asterisk.org/wiki/display/AST/SIP+Retransmissions).
-- Stopped music on hold on SIP/usuario7-00000001
-- Executing [h@internal:1] Hangup("SIP/usuario7-00000001", "") in new stack
== Spawn extension (internal, h, 1) exited non-zero on 'SIP/usuario7-00000001'
== Spawn extension (internal, 6000, 1) exited non-zero on 'SIP/usuario7-00000001'
== Using SIP RTP CoS mark 5
-- Executing [6021@internal:1] Dial("SIP/usuario0-00000003", "SIP/usuario7,30") in new stack
== Using SIP RTP CoS mark 5
-- Called SIP/usuario7
> 0x9def0b8 -- Probation passed - setting RTP source address to 192.168.1.75:7076
-- SIP/usuario7-00000004 answered SIP/usuario0-00000003
-- Remotely bridging SIP/usuario0-00000003 and SIP/usuario7-00000004
> 0x9def0b8 -- Probation passed - setting RTP source address to 192.168.1.75:7076
-- Executing [h@internal:1] Hangup("SIP/usuario0-00000003", "") in new stack
== Spawn extension (internal, h, 1) exited non-zero on 'SIP/usuario0-00000003'
== Spawn extension (internal, 6021, 1) exited non-zero on 'SIP/usuario0-00000003'
Asterisk*CLI>

```

Figura 7.3: Proceso de llamada en la consola de Asterisk.

Para simular 2 oficinas situadas en distintos lugares geográficamente se utilizaron 2 switches y 2 routers dentro del laboratorio de redes (G-302), se utilizaron 4 PC que fueron los usuarios que realizaban llamadas entre ellos a través de Asterisk, ya que estos usuarios fueron dados de alta previamente en el PBX.

Se realizaron distintas pruebas de en las que un usuario marcaba a una grabación del PBX, a otro usuario dentro y fuera de la red en la que se encontraba, para verificar el funcionamiento del servidor. Además se verifico que las características que se describieron funcionaran correctamente, a excepción de las video llamadas ya que no se contaban con cámaras web para poder realizar esta prueba.

El número de usuarios que soporta Asterisk depende del hardware con el que cuente el equipo donde esté instalado, ya que para una prueba tan pequeña con una PC personal fue suficiente.

No se pudo realizar las pruebas para realizar llamadas a números fijos ya que se necesita contar un servicio de Carrier con algún proveedor (Telmex, Axtel, Cablemas, etc.) para la conexión a un PSTN y una tarjeta analógica para la comunicación entre el Asterisk y el PSTN.

The screenshot shows the Wireshark interface with the following details for the selected packet (No. 11287):

- Ethernet II:** Src: Cisco\_a8:a0:6a (00:0f:34:a8:a0:6a), Dst: CadmusCo\_d8:4f:c2 (08:00:27:d8:4f:c2)
- Internet Protocol Version 4:** Src: 192.168.3.2 (192.168.3.2), Dst: 192.168.1.89 (192.168.1.89)
- User Datagram Protocol:** Src Port: sip (5060), Dst Port: sip (5060)
- Data (4 bytes):**

```

0000  08 00 27 d8 4f c2 00 0f 34 a8 a0 6a 08 00 45 00  ...'.O... 4..j..E.
0010  00 20 0a e9 00 00 7e 11 ac 38 c0 a8 03 02 c0 a8  .....~.8.....
0020  01 59 13 c4 13 c4 00 0c 38 8e 0d 0a 0d 0a 00 00  .Y..... 8.....
0030  00 00 00 00 00 00 00 00 00 00 00 00          .....

```

The status bar at the bottom indicates: File: "F:\captura" 2743 kB 00:17:29, Packets: 11287 · Displayed: 11287 (100.0%) · Load time: 0:00.828, Profile: Default.

Figura 7.4: Protocolo SIP en Wireshark.

The screenshot shows the Wireshark interface with a network capture of RTP traffic. The main pane displays a list of packets with columns for No., Time, Source, Destination, Protocol, Length, and Info. The selected packet (No. 7488) is highlighted in blue. The bottom pane shows the packet details for the selected RTP packet, including Ethernet II, Internet Protocol Version 4, and User Datagram Protocol. The hex dump pane shows the raw bytes of the packet.

No.	Time	Source	Destination	Protocol	Length	Info
7488	867.314340	192.168.1.7	192.168.1.89	RTCP	178	Sender Report source description
7505	867.468072	192.168.1.89	192.168.1.7	RTCP	106	Sender Report source description
8002	872.294230	192.168.1.7	192.168.1.89	RTCP	178	Sender Report source description
8020	872.468870	192.168.1.89	192.168.1.7	RTCP	106	Sender Report source description
8509	877.303926	192.168.1.7	192.168.1.89	RTCP	178	Sender Report source description
8527	877.468350	192.168.1.89	192.168.1.7	RTCP	106	Sender Report source description
9023	882.304374	192.168.1.7	192.168.1.89	RTCP	178	Sender Report source description
9040	882.468480	192.168.1.89	192.168.1.7	RTCP	106	Sender Report source description
9529	887.314304	192.168.1.7	192.168.1.89	RTCP	178	Sender Report source description
9545	887.467517	192.168.1.89	192.168.1.7	RTCP	106	Sender Report source description
10043	892.323911	192.168.1.7	192.168.1.89	RTCP	178	Sender Report source description
10059	892.467982	192.168.1.89	192.168.1.7	RTCP	106	Sender Report source description
10918	954.196510	192.168.1.7	192.168.1.89	RTCP	178	Sender Report source description
390	157.103954	192.168.3.2	192.168.1.89	RTP	62	Unknown RTP version 0
391	157.104159	192.168.1.89	192.168.3.2	RTP	74	Unknown RTP version 0
392	157.104342	192.168.1.89	192.168.3.2	RTP	84	Unknown RTP version 0
393	157.119019	192.168.3.2	192.168.1.89	RTP	62	Unknown RTP version 0
395	157.121000	192.168.1.89	192.168.3.2	RTP	74	Unknown RTP version 0
396	157.121001	192.168.1.89	192.168.3.2	RTP	84	Unknown RTP version 0
397	157.133804	192.168.3.2	192.168.1.89	RTP	214	PT=ITU-T G.711 PCMU, SSRC=0x5044, Seq=0, Time=80
398	157.133983	192.168.1.89	192.168.1.3	RTP	214	PT=ITU-T G.711 PCMU, SSRC=0x5044, Seq=0, Time=80, Mark
399	157.148269	192.168.3.2	192.168.1.89	RTP	214	PT=ITU-T G.711 PCMU, SSRC=0x5044, Seq=1, Time=240
400	157.148425	192.168.1.89	192.168.1.3	RTP	214	PT=ITU-T G.711 PCMU, SSRC=0x5044, Seq=1, Time=240
401	157.150871	192.168.1.3	192.168.1.89	RTP	214	PT=ITU-T G.711 PCMU, SSRC=0x251E7B4F, Seq=2, Time=800
402	157.150956	192.168.1.89	192.168.3.2	RTP	214	PT=ITU-T G.711 PCMU, SSRC=0x251E7B4F, Seq=2, Time=800, M
403	157.159077	192.168.1.3	192.168.1.89	RTP	214	PT=ITU-T G.711 PCMU, SSRC=0x251E7B4F, Seq=3, Time=960

Frame 7488: 178 bytes on wire (1424 bits), 178 bytes captured (1424 bits) on interface 0  
 Ethernet II, Src: HewlettP\_5a:78:23 (28:92:14:5a:78:23), Dst: CadmusCo\_d8:4f:c2 (08:00:27:d8:4f:c2)  
 Internet Protocol Version 4, Src: 192.168.1.7 (192.168.1.7), Dst: 192.168.1.89 (192.168.1.89)  
 User Datagram Protocol, Src Port: 7079 (7079), Dst Port: 12767 (12767)  
 Real-time Transport Control Protocol (Sender Report)

```

0000 08 00 27 d8 4f c2 28 92 4a 5a 78 23 08 00 43 00  ..O(.JZx#..E.
0010 00 a4 9b 82 40 00 40 11 1b 16 c0 a8 01 07 c0 a8  ....@. ....
0020 01 59 1b a7 31 df 00 90 af ee 81 c8 00 0c 01 32  .Y..1... ..2
0030 09 40 d7 8a 2c 8f e4 51 c9 f7 00 02 6c f0 00 00  .@....Q ....l...
0040 03 fb 00 02 53 70 2d 90 a9 42 00 00 00 00 00 00  ...Sp-. .B.....
0050 93 29 00 00 00 02 2c 89 78 8c 00 04 d7 26 81 ca  .).....X....&..
  
```

File: Fillophora\_274318.00:17:29 Packets: 11287 · Deployed: 11287 (100.0%) · Load time: 0:00:812 Profile: Default

Figura 7.5: Protocolo RTP en Wireshark.

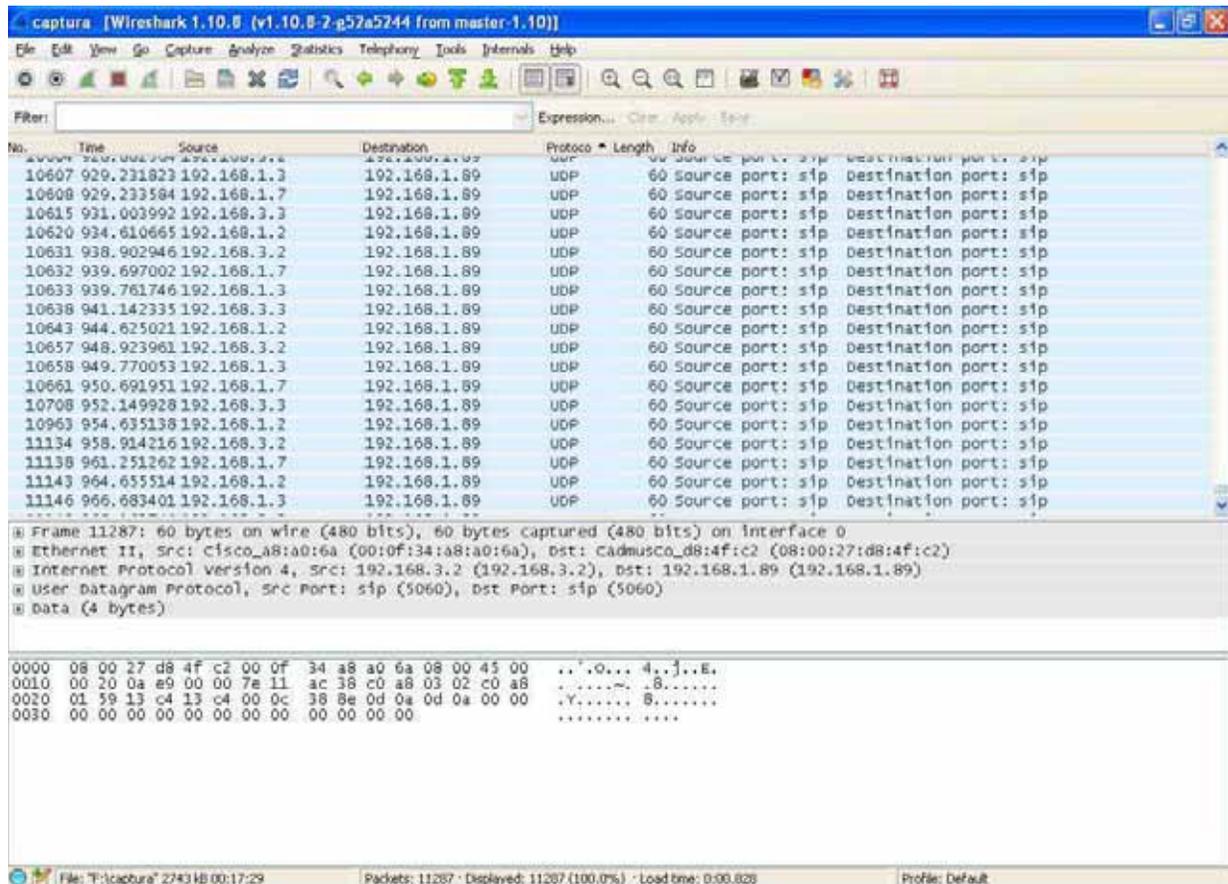


Figura 7.6: Protocolo UDP en Wireshark.

**Terminal (como superusuario)**

Archivo Editar Ver Buscar Terminal Pestañas Ayuda

Terminal		Terminal	
05675.0	1407105675.0		
2014-08-03 17:58:48	usuario0	usuario0	6021
Dial	SIP/usuario7,30	0	15
06728.2	1407106728.2		
2014-08-03 18:17:13	usuario7	usuario7	6000
Dial	SIP/usuario0,30	0	74
07833.4	1407107833.4		
2014-08-03 19:13:27	usuario0	usuario0	6021
VoiceMail	7021@default	0	4
11207.6	1407111207.6		
2014-08-03 19:14:11	usuario0	usuario0	6021
Dial	SIP/usuario7,30	0	62
11251.7	1407111251.7		
2014-08-03 19:21:21	usuario0	usuario0	6021
Dial	SIP/usuario7,30	0	16
11681.9	1407111681.9		
2014-08-03 19:33:23	usuario0	usuario0	6021
VoiceMail	7021@default	0	3
12403.11	1407112403.11		
2014-08-03 19:33:31	usuario0	usuario0	6021
Dial	SIP/usuario7,30	0	12
12411.12	1407112411.12		
2014-08-03 19:48:41	usuario0	usuario0	6021
Dial	SIP/usuario7,30	0	10
13321.14	1407113321.14		
2014-08-03 19:49:58	usuario0	usuario0	6021
Dial	SIP/usuario7,30	0	10
13398.16	1407113398.16		
2014-08-03 19:50:56	usuario0	usuario0	6021
Dial	SIP/usuario7,30	0	11
13456.18	1407113456.18		
2014-08-03 19:56:28	usuario7	usuario7	6000
Dial	SIP/usuario0,30	0	91
13788.0	1407113788.0		

Figura 7.7: Protocolo UDP en Wireshark.



# Bibliografía

- [1] Wen-mei W. Hwu David B. Kirk. Programming massively parallel processors a hands-on approach. *Editorial Morgan Kaufmann*, 81:1451–1465, 1993.
- [2] José M. Huidobro. Comunicaciones en redes wlan. *Editorial Linusa*, 2006.
- [3] Asterisk. [www.asterik.org](http://www.asterik.org), Consultada en mayo del 2014.
- [4] Linphone. [www.linphone.org](http://www.linphone.org), Consultada en mayo del 2014.
- [5] Jim Van Meggelen. Asterisk: The future of thelephony. *Editorial O'Reilly*, 2007.
- [6] Rfc3261-sip: Session initiation protocol. [www.fags.org/rfc3261.html](http://www.fags.org/rfc3261.html), Consultada en mayo del 2014.
- [7] Curso sobre voip y asterisk v1. irontec. [www.irontec.com](http://www.irontec.com), Consultada en junio del 2014.
- [8] Wikiasterisk. [www.wikiasterisk.org](http://www.wikiasterisk.org), Consultada en mayo del 2014.
- [9] David Escanuela Alonso. Estudio y montaje a medida de centralita telefónica voip asterisk. 2010. España.



# Apéndice A

## Archivos de Configuración de Asterisk

---

### Archivo sip.conf

```
1 [general]
2 bindaddr=0.0.0.0
3 udpbindaddr=0.0.0.0
4 tcpenable=no
5 tcpbindaddr=0.0.0.0
6 transport=udp
7
8
9 context=default
10 videosupport=yes
11
12 allow=g729
13 allow=h261
14 allow=h263
15 allow=h263p
16 allow=h264
17
18 ;usuarios dados de alta en asterisk
19 [usuario0]
20 type=friend
21 context=internal
22 host=dynamic
23 secret=uam
24 disallow=all
25 allow=ulaw
26 mailbox=7000@default
27 callgroup=4
28 pickupgroup=1,2,3,4
29 cc_agent_policy=generic
30 cc_monitor_policy=generic
31
32 [usuario1]
33 type=friend
34 context=internal
35 host=dynamic
36 secret=uam
37 disallow=all
38 allow=ulaw
39 mailbox=7001@default
40 callgroup=1
41 pickupgroup=1
42
43 [usuario2]
44 type=friend
45 context=internal
46 host=dynamic
47 secret=uam
48 disallow=all
49 allow=ulaw
50 mailbox=7002@default
51 callgroup=1
```

```
52 pickupgroup=1
53
54 [usuario3]
55 type=friend
56 context=internal
57 host=dynamic
58 secret=uam
59 disallow=all
60 allow=ulaw
61 mailbox=7003@default
62 callgroup=1
63 pickupgroup=1
64
65 [usuario4]
66 type=friend
67 context=internal
68 host=dynamic
69 secret=uam
70 disallow=all
71 allow=ulaw
72 mailbox=7011@default
73 callgroup=2
74 pickupgroup=2
75
76 [usuario5]
77 type=friend
78 context=internal
79 host=dynamic
80 secret=uam
81 disallow=all
82 allow=ulaw
83 mailbox=7012@default
84 callgroup=2
85 pickupgroup=2
86
87 [usuario6]
88 type=friend
89 context=internal
90 host=dynamic
91 secret=uam
92 disallow=all
93 allow=ulaw
94 mailbox=7013@default
95 callgroup=2
96 pickupgroup=2
97
98 [usuario7]
99 type=friend
100 context=internal
101 host=dynamic
102 secret=uam
103 disallow=all
104 allow=ulaw
105 mailbox=7021@default
106 callgroup=3
107 pickupgroup=3
108
109 [usuario8]
110 type=friend
111 context=internal
112 host=dynamic
113 secret=uam
114 disallow=all
115 allow=ulaw
116 mailbox=7022@default
117 callgroup=3
118 pickupgroup=3
119
120 [usuario9]
121 type=friend
122 context=internal
123 host=dynamic
124 secret=uam
125 disallow=all
126 allow=ulaw
127 mailbox=7023@default
128 callgroup=3
129 pickupgroup=3
```

asterisk/sip.conf

## Archivo extensions.conf

```

1 [internal]
2 exten => 6000,1,Dial(SIP/usuario0,30)
3 exten => 99,1,CallCompletionRequest()
4 exten => 99,n,Hangup()
5 exten => 98,1,CallCompletionCancel()
6 exten => 98,n,Hangup()
7 exten => 6000,n,VoiceMail(7000@default)
8 exten => 6000,n,MusicOnHold(default)
9
10 exten => 6001,1,Dial(SIP/usuario1,30)
11 exten => 6001,n,VoiceMail(7001@default)
12 exten => 6001,n,MusicOnHold(default)
13
14 exten => 6002,1,Dial(SIP/usuario2,30)
15 exten => 6002,n,VoiceMail(7002@default)
16 exten => 6002,n,MusicOnHold(default)
17
18 exten => 6003,1,Dial(SIP/usuario3,30)
19 exten => 6003,n,VoiceMail(7003@default)
20 exten => 6003,n,MusicOnHold(default)
21
22 exten => 6011,1,Dial(SIP/usuario4,30)
23 exten => 6011,n,VoiceMail(7011@default)
24 exten => 6011,n,MusicOnHold(default)
25
26 exten => 6012,1,Dial(SIP/usuario5,30)
27 exten => 6012,n,VoiceMail(7012@default)
28 exten => 6012,n,MusicOnHold(default)
29
30 exten => 6013,1,Dial(SIP/usuario6,30)
31 exten => 6013,n,VoiceMail(7013@default)
32 exten => 6013,n,MusicOnHold(default)
33
34 exten => 6021,1,Dial(SIP/usuario7,30)
35 exten => 6021,n,VoiceMail(7021@default)
36 exten => 6021,n,MusicOnHold(default)
37
38 exten => 6022,1,Dial(SIP/usuario8,30)
39 exten => 6022,n,VoiceMail(7022@default)
40 exten => 6022,n,MusicOnHold(default)
41
42 exten => 6023,1,Dial(SIP/usuario9,30)
43 exten => 6023,n,VoiceMail(7023@default)
44 exten => 6023,n,MusicOnHold(default)
45
46 include => aplicaciones
47
48
49 [aplicaciones]
50 ;Hola Mundo
51 exten => 1000,1,Answer()
52 exten => 1000,2,Wait(1)
53 exten => 1000,3,Playback(hello-world)
54 exten => 1000,4,Hangup()
55
56 ;Correo de Voz
57 exten => 1100,1,Answer()
58 exten => 1100,n,VoiceMailMain(@default)
59
60 ;captura de llamadas
61 exten => *8XXXX,1,Pickup(${EXTEN:2})
62 same => n,Hangup()
63
64 ;Conferencias
65 exten => 1200,1,Goto(conf,1)
66 exten => conf,1,Set(MEETME_RECORDINGFILE=/tmp/Tutorial-${TIMESTAMP})
67 exten => conf,2,Meetme(8000,M)
68 exten => conf,3,Hangup()
69
70 exten => 1210,1,Goto(conf,1)
71 exten => conf,1,Set(MEETME_RECORDINGFILE=/tmp/Tutorial-${TIMESTAMP})
72 exten => conf,2,Meetme(8001,M)
73 exten => conf,3,Hangup()
74
75 ;musuca en espera
76 exten => 2000,1,Answer()
77 exten => 2000,n,MusicOnHold(default)
78 exten => 2000,n,PlayBack(welcome)
79 exten => 2000,n,Hangup()
80
81
82 ;DND
83 exten => 99,1,Macro(DND)
84 exten => 99,n,Hangup()
85

```

```

86 exten => _6XXX,1,GotoIf(${DB_EXISTS(DND/${EXTEN})})?DND-ON)
87 exten => _6XXX,n,Dial(SIP/${EXTEN},20,t); DND esta en off, inicia el dialing
88 exten => _6XXX,n,GotoIf("${DIALSTATUS}"="BUSY"]?busy:unavail)
89 exten => _6XXX,n(busy),Voicemail(${EXTEN}@default,b)
90 exten => _6XXX,n,Hangup()
91 exten => _6XXX,n(unavail),Voicemail(${EXTEN}@default,u)
92 exten => _6XXX,n,Hangup()
93 exten => _6XXX,n(DND-ON),GotoIf(2100,6)
94 exten => h,1,Hangup()
95
96 [macro-DND]
97 exten => s,1,GotoIf(${DB_EXISTS(DND/${CALLERID(num)})})?disable-dnd,1)
98 exten => s,n,Set(DB(DND/${CALLERID(num)})=1)
99 exten => s,n,PlayBack(do-not-disturb)
100 exten => s,n,PlayBack(activated)
101 exten => s,n,Hangup()
102 exten => disable-dnd,1,Set(DEL=${DB_DELETE(DND/${CALLERID(num)})})
103 exten => disable-dnd,n,PlayBack(do-not-disturb)
104 exten => disable-dnd,n,PlayBack(de-activated)
105 exten => disable-dnd,n,Hangup()
106
107 ;calendario
108 [calendario]
109 exten => cal,1,NoOp(Llamada desde el calendario)
110 same => n,Agi(googletts.agi,"${CALENDAR_EVENT(summary)}.",es) ;contiene titulo del evento
111 same => n,Agi(googletts.agi,"a las ${STRFTIME($CALENDAR_EVENT(start))},,%H:%M).",es) ;contiene la hora del
112     evento el formato de la hora esta en EPOCH hay que convertirla a hora y min con STRFTIME
113 same => n,Wait(1)
114 same => n,PalyBack(goodbye)
115 same => n, Hangup()

```

asterisk/extensions.conf

## Archivo voicemail.conf

```

1 [default]
2 7000=>7000,Usuario0,usuario@dominio.com
3
4 7001=>7001,Usuario1,usuario@dominio.com
5 7002=>7002,Usuario2,usuario@dominio.com
6 7003=>7003,Usuario3,usuario@dominio.com
7
8 7011=>7011,Usuario4,usuario@dominio.com
9 7012=>7012,Usuario5,usuario@dominio.com
10 7013=>7013,Usuario6,usuario@dominio.com
11
12 7021=>7021,Usuario7,usuario@dominio.com
13 7022=>7022,Usuario8,usuario@dominio.com
14 7023=>7023,Usuario9,usuario@dominio.com

```

asterisk/voicemail.conf

## Archivo features.conf

```

1
2 [general]
3 parkext => 700
4 parkpos => 701-720
5 context => parkedcalls
6 parkingtime => 45
7 findslot => next
8 parkedmusicclass=default
9 transferdigittimeout => 3
10 xfersound = beep
11 xferfailsound = beeper
12 pickupexten = *8
13 atxfernoanswertimeout = 15
14
15 [featuremap]
16 blindxfer => ##
17 atxfer => *2
18 disconnect => *
19 automon => *1

```

asterisk/features.conf

## Archivo meetme.conf

```

1 [general]
2 audiobuffers=32
3 schedule=yes
4 logmembercount=yes
5 fuzzystart=300
6 earlyalert=3600
7 endalert=120
8
9
10 [rooms]
11
12 conf => 8000
13 conf => 8001,1234,5678

```

asterisk/meetme.conf

## Archivo calendar.conf

```

1
2 [calendario]
3 type = caldav
4 url = https://www.google.com/calendar/dav/usuario@dominio.com/events
5 user = usuario@dominio.com
6 secret = password
7 refresh = 15
8 timeframe = 180
9 autorereminder=10
10 chanel=SIP/6001
11 context=calendario
12 extencion=cal
13 waittime=45

```

asterisk/calendar.conf

## Archivo musichold.conf

```

1 [general]
2 cachertclasses=yes
3
4 [default]
5 mode=files
6 directory=/var/lib/asterisk/mohmp3
7 random=yes
8 announcement=queue-thankyou

```

asterisk/musichold.conf

## Archivo cdr\_mysql.conf

```

1 [global]
2 hostname=localhost
3 dbname=asteriskCDR
4 table=cdr
5 password=password
6 user=root
7 port=3306
8 sock=/var/run/mysqld/mysqld.sock
9
10 [columns]
11 ;static "" =>
12 ;alias =>
13 alias start => calldate
14 ;alias cllerid => clid
15 ;alias src => src
16 ;alias dst => dst
17 ;alias dcontext => dcontext
18 ;alias channel => channel
19 ;alias dstchannel => dstchannel
20 ;alias lastapp => lastapp

```

```
21 ;alias lastdata => lastdata
22 ;alias duration => duration
23 ;alias billsec => billsec
24 ;alias disposition => disposition
25 ;alias amaflags => amaflags
26 ;alias accountcode => accountcode
27 ;alias userfield => userfield
28 ;alias uniqueid => uniqueid
```

asterisk/cdr\_mysql.conf