



**UNIVERSIDAD NACIONAL AUTÓNOMA
DE MÉXICO**

**FACULTAD DE ESTUDIOS SUPERIORES
ARAGÓN**

**“IMPLEMENTACIÓN DE ASTERISK-VoIP
COMO ALTERNATIVA LIBRE DE
COMUNICACIÓN INTERNA PARA LUZ Y
FUERZA DEL CENTRO.”**

TESIS

**PARA OBTENER EL TÍTULO DE
INGENIERO EN COMPUTACIÓN**

PRESENTA:

ANGEL DANIEL PEREGRINO JUAREZ



**ASESOR: M. EN C. JESÚS HERNANDEZ CABRERA
MARZO 2011**



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

AGRADECIMIENTOS

A mi madre, por el apoyo incondicional que me dio a lo largo de la carrera y de mi vida.

A mis hermanos por el apoyo que me brindaron.

A mi tío que fue parte importante y apoyo para mi carrera.

A la UNAM, por brindarme los conocimientos, las experiencias y los valores morales en los cuales me sustentó.

A mis amigos y compañeros “Los cinco de siempre” por su amistad y compañerismo.

A mi asesor por sus aportaciones y paciencia en el desarrollo de esta tesis.

A mi segundo asesor por sus aportaciones y apoyo para la realización de esta tesis.

IMPLEMENTACIÓN DE ASTERISK-VoIP COMO ALTERNATIVA LIBRE DE COMUNICACIÓN INTERNA PARA LUZ Y FUERZA DEL CENTRO.

CAPÍTULO 1: ANTECEDENTES DE LA TECNOLOGÍA VoIP

1.1.- Tecnología VoIP.	2
1.1.1.- Antecedentes de VoIP	2
1.1.2.- ¿Qué es VoIP?	5
1.1.3.- ¿Cómo funciona?	6
1.1.4.- Estándares y protocolos.	6
1.1.4.1.-Protocolo H323.	6
1.1.4.2.-Protocolo H248(Megaco).	7
1.1.4.3.-Protocolo IAX.	8
1.1.4.4.-Protocolo SIP.	8
1.1.5.- ¿Por qué la Tecnología VoIP?	8
1.1.6.- ¿Donde se puede integrar esta Tecnología?	9
1.1.7.- ¿Por qué en Luz y Fuerza?	9
1.2.-Alternativas.	10
1.2.1. - Microsoft Office Communication Server 2007	10
1.2.2.- openPBX	10
1.2.3.- 3CX	11
1.2.4.- SIPX	11
1.2.5.- Asterisk	12

CAPÍTULO 2.- ANÁLISIS DE FACTIBILIDAD PARA LA IMPLEMENTACIÓN DE TELEFONÍA VoIP (ASTERISK) EN LYF.

2.1.- Asterisk.	14
2.1.1.- ¿Qué es Asterisk? .	14
2.1.2.- ¿El porque de Asterisk?	15
2.1.3.- Antecedentes de Asterisk.	16
2.1.4.- Ventajas.	17
2.1.5.- Sistemas Operativos Soportados.	18
2.2.-Análisis de Factibilidad.	19
2.2.1.- Comparación de Costos.	19
2.2.2.- Reutilización e infraestructura.	19
2.2.3.- Factibilidad de uso.	20
2.2.4.- Licencias.	22
2.2.5.- Capacitación de usuarios finales.	24
2.2.6.- Conclusión de la factibilidad.	25

CAPÍTULO 3: INTRODUCCIÓN A LA TECNOLOGÍA VoIP.

3.1.- Introducción.	27
3.1.1.-Telefonía tradicional.	28
3.1.1.1.-Historia.	28

3.1.1.2.-PSTN .	35
3.1.1.3.-RDSI ó ISDN.	35
3.1.2.-Telefonía IP.	36
3.1.2.1.-Orígenes.	36
3.1.2.2.-¿Como funciona la tecnología IP?.	38
3.1.2.3.-Estándares y Protocolos de señalización IP.	45
3.1.2.4.-Alternativas.	46
3.1.2.4.1.-H323.	46
3.1.2.4.2.-MGCP.	48
3.1.2.4.3.-MEGACO.	49
3.1.2.4.4.-SIP.	50
3.1.2.4.5.-IAX2.	51
3.1.3- Ventajas de protocolos.	52
3.1.4.- Protocolo elegido para la implementación en LYF (¿Por qué?).	53
3.1.5.- Protocolo SIP.	53
3.1.5.1.-Diseño de protocolo SIP.	55
3.1.5.2.-Funcionamiento del protocolo SIP.	56
3.1.5.2.1.- Agentes de Usuario.	57
3.1.5.2.2.-Servidores de Registro.	58
3.1.5.2.3.-Servidores Proxy.	59
3.1.5.2.4.-Servidores de Redirección.	60
3.1.5.2.5.-Formato de los mensajes.	60
3.1.5.2.6.-Métodos (Solicitudes).	61
3.1.5.2.7.- Códigos de Estado (Respuestas).	61
3.1.5.2.8.-Flujo de establecimiento de una sesión.	62
3.2.-Calidad del Servicio(QoS)	63
3.2.1.-Factores que impactan la calidad del servicio.	63
3.2.1.1.-Latencia.	64
3.2.1.2.-Jitter.	64
3.2.1.3.-Eco.	64
3.2.1.4.-Perdida de Paquetes.	65
3.2.2.-Conversión analógico-digital.	65
3.2.2.1.-Voz sobre Frame Relay.	66
3.2.2.2.-Voz sobre ATM.	66
3.2.2.3.-Voz sobre IP.	67
3.2.3.-Codificadores	67
3.3.-Componentes.	68
3.3.1.- Terminales Voip(Teléfonos IP).	68
3.3.2.- Terminales Voip(Softphone).	69
3.4.-Beneficios de implantar VoIP.	69

CAPÍTULO 4: INSTALACIÓN Y CONFIGURACIÓN DE ASTERISK (SERVIDOR)

4.1.- Asterisk.	71
------------------------	----

4.1.1.- ¿Cómo Funciona?.	71
4.1.2.- Componentes de Asterisk.	71
4.1.3.- Software necesario.	73
4.1.3.1.-Sistema Operativo.	74
4.1.3.2.- Paquetes(software necesario).	75
4.1.3.3.- Librerías.	75
4.2.- Instalación del Sistema Operativo.	76
4.3.- Instalación de Asterisk.	79
4.4.- Instalación de Asterisk-gui.	83
4.5.- Directorios de instalacion de Asterisk.	86
4.6.-Canal.	88
4.7.-Gramática de Asterisk.	88
4.7.1.-Gramática de grupo simple.	89
4.7.2.-Gramática de Herencia de Opciones.	90
4.7.3.-Gramática de Entidad Compleja.	90
4.8.-Configuración de Asterisk.	91
4.8.1.- Configuración de sip.conf.	91
4.8.2.- DialPlan (Plan de Marcado).	96
4.8.2.1.- Extensiones.	97
4.8.2.2.- Aplicaciones.	98
4.8.2.3.- Contexto.	100
4.8.2.4.- Prioridad.	100
4.8.3.-Dial.	101
4.8.4.-Configuración de extensions.conf.	104
4.9.- Administración básica del CLI(command line interface).	106
4.10.-Verificación de servicios vía consola.	107

CAPÍTULO 5: INSTALACIÓN Y CONFIGURACIÓN DE SOFTPHONE PARA ASTERISK (CLIENTE).

5.1.- Sotfphone	109
5.1.1.- ¿Qué son?	109
5.1.2.- ¿Por qué Softphone para el proyecto?	111
5.2.- Instalación de Xlite 3.x	112
5.3.- Configuración de Xlite	113
5.4.- Verificación de intercomunicación entre clientes.	113
 CONCLUSIONES	 117
BIBLIOGRAFIA	120

INDICE DE TABLAS.

CAPÍTULO 1: ANTECEDENTES DE LA TECNOLOGÍA VoIP.

1.1.- Características de H323.	7
1.2.- Características de IAX.	8
1.3.- Beneficios 3CX.	11
1.4.- Características de Asterisk.	12

CAPÍTULO 2.- ANÁLISIS DE FACTIBILIDAD PARA LA IMPLEMENTACIÓN DE TELEFONÍA VoIP (ASTERISK) EN LYF

2.1.- Líneas de código de Asterisk.	14
2.2.- Comparación de costos.	19
2.3.- Comparación de costos de Software.	24

CAPÍTULO 3: INTRODUCCIÓN A LA TECNOLOGÍA VoIP.

3.1.- Características de codec's para la conversión analógico-digital.	44
3.2.- Comparación de ventajas.	52
3.3.- Características y funciones del protocolo SIP.	55
3.4.- Métodos (Solicitudes).	61
3.5.- Códigos de estado (respuestas).	61
3.6.- Codec's populares.	68

CAPÍTULO 4: INTALACIÓN Y CONFIGURACIÓN DE ASTERISK (SERVIDOR).

4.1.- Distribuciones.	74
4.2.- Paquetes necesarios.	75
4.3.- Librerías necesarias.	75
4.4.- Instalación de Debian.	76-79
4.5.- Directorios de Instalación.	87
4.6.- Archivos de configuración para los canales.	88
4.7.- Significado de los signos.	89
4.8.- Tipos de gramáticas.	89
4.9.- Opciones para la sección global "[general]"	92
4.10.- Parametros para sip.conf.	93
4.11.- Aplicaciones Generales.	98
4.12.- Aplicaciones de gestión de llamadas.	98
4.13.- Aplicaciones de control de flujo.	99
4.14.- Aplicaciones de reproducción de sonidos.	99

CAPÍTULO 5: INSTALACIÓN Y CONFIGURACIÓN DE SOFTPHONES PARA WINDOWS (Clientes).

5.1.- Características de la llamada IP.	114
---	-----

INDICE DE IMÁGENES.

CAPÍTULO 1: ANTECEDENTES DE LA TECNOLOGÍA VoIP.

1.1.- PMBX.	2
1.2.- VoIP.	5
1.3.- Tecnología VoIP.	9
1.4.- Logo LYF.	9
1.5.- Office Communications.	10
1.6.- Open PBX.	10
1.7.- 3CX.	11

CAPÍTULO 2.- ANÁLISIS DE FACTIBILIDAD PARA LA IMPLEMENTACIÓN DE TELEFONÍA VoIP (ASTERISK) EN LYF

2.1.-Logo Asterisk	14
2.2.-Mark Spencer(foto)	16
2.3.- GNU	22
2.4.- Licencia GPL2	24

CAPÍTULO 3: INTRODUCCIÓN A LA TECNOLOGÍA VoIP.

3.1.-Telégrafo.	28
3.2.-Alejandro Graham Bell.	28
3.3.-Antonio Meucci.	29
3.4.-Guglielmo Marconi.	30
3.5.- Mapa cableado submarino.	31
3.6.-Dr. Martin Cooper.	34
3.7.-PSTN (Public Switched Telephone Network).	35
3.8.-RDSI (Red Digital de Servicios Integrados) o ISDN (Ingles).	35
3.9.-Conexión de teléfono.	36
3.10.-Señal analógica.	39
3.11.-Señal digital.	40
3.12.-Cuantificación uniforme.	41
3.13.-Transformación analógico-digital.	42
3.14.-Cuantificación de la señal analógica.	43
3.15.-Transformación de la cuantificación a binario.	43
3.16.-Arquitectura H323.	47
3.17.-Esquema del protocolo MEGACO.	49
3.18.-Esquema de conexión IAX2.	51
3.19.-Esquema de funcionamiento del protocolo SIP.	56
3.20.-Funcionamiento de los servidores de registro.	58
3.21.-Ejemplo de servidores proxy.	59
3.22.-Flujo de sesión.	62
3.23.-Dialogo (peer to peer).	62
3.24.-Teléfono IP.	68
3.25.-Softphone.	69

CAPÍTULO 4: INTALACIÓN Y CONFIGURACIÓN DE ASTERISK (SERVIDOR).

4.1.-Esquema de los componentes de Asterisk.	73
4.2.-Instalación de Debian.	76
4.3.-Idioma instalación.	76
4.4.-Configuración teclado.	77
4.5.-Configuración de red.	77
4.6.-Configuración de dominio.	77
4.7.-Configuración reloj.	78
4.8.-Configuración usuarios.	78
4.9.-Paquetería.	78
4.10.-Instalación GRUP.	79
4.11.-Instalación de gcc.	79
4.12.-Instalación g++.	80
4.13.-Instalación make.	80
4.14.-Instalación de Linux Headers.	80
4.15.-Descarga de Fuentes (Asterisk).	81
4.16.-Logo Asterisk.	82
4.17.-Instalación de Asterisk.	82
4.18.-Fin de Instalación.	82
4.19.-Vista de procesos.	86
4.20.-Directorios de instalación.	87
4.21.-Arquitectura del DialPlan.	97
4.22.-Flujo de la llamada con aplicación Dial.	101
4.23.-Verificación vía consola de usuarios.	107
4.24.-Verificación de los usuarios OnLine.	107

CAPÍTULO 5: INSTALACIÓN Y CONFIGURACIÓN DE SOFTPHONES PARA WINDOWS (Clientes).

5.1.-Softphone WengoPhone.	109
5.2.- Softphone Gizmoes.	110
5.3.- Softphone SighSpeed.	110
5.4.- Softphone Xlite.	110
5.5.- Softphone SJPHane.	111
5.6.- Softphone Zoiper.	111
5.7.-Instalación de Xlite.	112
5.8.-Pantalla de Xlite al estar registrado correctamente.	113
5.9.- Pantalla de Xlite al llamar entre extensiones.	113
5.10.-Llamada de voz IP (caso1).	114
5.11.-Características de la llamada IP.	114
5.12.-Desglose de paquetes enviados (caso1).	115
5.13.-Flujo del establecimiento de una sesión (caso1).	115
5.14.-Características de la llamada de voz IP (caso2).	116
5.15.-Estadística SIP.	116

CAPÍTULO 1

ANTECEDENTES DE LA TECNOLOGÍA VoIP

1.1.- Tecnología VoIP

1.1.1.- Antecedentes de VoIP

A principios de siglo las redes públicas empezaron como PMBX (Private Manual Branch EXchange) en las cuales las llamadas eran transferidas de un circuito a otro manualmente, se conectaban cables entre los abonados utilizando personal para estas labores, en esa época los teléfonos no tenían ruedas o teclados para marcar el numero de destino, para "marcar" descolgabas el teléfono y le decías a la operadora el abonado de destino o en las ciudades más grandes el numero del abonado de destino.



Figura 1. 1.- PMBX.

Este método generó innumerables problemas pero hubo dos muy grandes los cuales provocaron cambios drásticos debido a “errores” humanos lo cual provocaron una evolución.

El primero era que los operadores que en su mayoría eran mujeres y niños escuchaban las llamadas. La segunda y más importante era la monopolización de empresas la cual consistía que en cuanto descolgaban los usuarios y pedían que los comunicaran a algún negocio específico los operadores los comunicaban ciertamente a un negocio de igual giro pero diferente dueño. Para resolver esto la única manera es que fuera realizado por máquinas automáticas para así hacer que no existiera el “error” humano.

Así nacieron las primeras centrales públicas electromecánicas, estas eran muy costosas pero eran mucho más eficientes y por no tener tanto personal, no era tan costosa su operación, y para las ciudades eran rentables y podían funcionar las 24 horas, mientras esto ocurría, la electrónica fue avanzando, para permitir que estas centrales fueran electrónicas, con esto consumían mucho menos energía, costaban menos y eran mucho más pequeñas.

En este punto el desarrollo de la telefonía empezó a estar ligado del desarrollo de la computación, y aunque estos nuevos equipos todavía utilizan tecnología convencional su núcleo empezó a ser un computador. Esto empezó a aumentar las posibilidades de agregar funcionalidades ya que es más fácil, rápido y económico, desarrollar software que desarrollar componentes electrónicos.

Por otro lado la gente de la industria de los datos, comenzó a experimentar con diferentes tecnologías para lograr desarrollar una única red convergente que se encargase de transportar todo tipo de comunicación, ya sea voz, datos, vídeo o cualquier tipo de información, para lo cual desarrollaron una tecnología encargada de encapsular la voz en paquetes para poder ser transportada en redes de datos y así nació la VoIP.

El crecimiento y desarrollo de técnicas avanzadas de digitalización de voz, así como los mecanismos de control y priorización de tráfico, protocolos de transmisión en tiempo real y el estudio de nuevos estándares que garantizan la calidad de servicio en redes IP, han hecho posible un entorno donde se puede transmitir voz sobre IP.

La red de telefonía mundial fue diseñada para reproducir con claridad voces humanas, para realizarlo utiliza un sistema que es capaz de transmitir señales entre 350Hz y 3400Hz. La conversión de estas señales *análogas* a *digitales* es llamada PCM ("Pulse Code Modulation").¹

VoIP comenzó como el resultado del trabajo de un grupo de jóvenes en Israel durante 1995. Al principio del proyecto la única comunicación era de PC a PC por medio de IP. Poco más tarde Vocaltec,inc. Anuncio el lanzamiento del primer Softphone que llamaron "Internet Phone Software". Este Softphone estaba diseñado para ser usado en dos PC's con las mismas características, deberían tener: tarjeta de sonido, micrófono, bocinas y modem. Este fue comercialmente un fracaso ya que la velocidad de conexión era muy baja y las comunicaciones por medio de banda ancha aun no estaban disponibles.²

¹ Conectividad VoIP, http://www.osmosislatina.com/conectividad/tipos_de_conexion.htm#cswitched [Consulta: Diciembre 2009]
² Denise Donohue, "Voice Gateways and Gatekeeper", Cap 9 Voice Over IP, Prentice Hall, 1999.

En 1997 Jeff Pulver decide juntar a los pocos usuarios, fabricantes e interesados en esta tecnología, el primer congreso y que actualmente es el más grande evento de VoIP. Pulver también formo una compañía prestadora de servicios VoIP llamada FreeWorldDialup comúnmente llamada FWD y es co-fundador de Vonage, el proveedor de VoIP mas grande de EEUU.

En 1998 Voip dio un gran salto. Un grupo de emprendedores comenzó a fabricar los primeros ATA y Voip-Gateways para permitir las primeras comunicaciones PC-a-Tel. convencional y finalmente las primeras comunicaciones de Tel- convencional-a-Tel. convencional con un adaptador ATA en cada Extremo. Al inicio este servicio no tenia cargo alguno para que pudieran probar la calidad y la tecnología. Estas llamadas contenían publicidad al inicio y al final de la comunicación. En este punto Voip sumaba el 1% del total del tráfico de Voz. Durante 1998 tres fabricantes comenzaron a fabricar switches de capa 3 con QoS.³

En 1999, cisco vende sus primeras plataformas corporativas para VoIP. Se utilizaba principalmente el protocolo H.323 de señalizacion.⁴

En el año 2000 VoIP representaba más del 3% del tráfico de voz. El mismo año Mark Spencer un estudiante de la universidad de Auburn crea Asterisk, la primer central telefónica conmutador basado en Linux con una PC personal con un código fuente abierto.⁵

En 2002 el protocolo SIP comienza a desplazar al H.323.

En 2003 dos jóvenes Jan Friis y Niklass Zenntrom crean un softphone gratuito llamado Skype, el cual se propago de una manera descomunal contando para el 2005 con 50 millones de usuarios de todo el mundo.

³ CAVE, Hellis. Conferencia Magistral Simposium Internacional de Telematica Inter Voice Brite Inc. 2001.

⁴ Historia de VoIP, <http://voipex.blogspot.com/2006/04/historia-de-voip.html>, [Consulta: Diciembre 2009]

⁵ Osland P. y Dinh, K. Perceived Voip quality under varying traffic conditions, 17th Nordic Teletraffic Seminar, Norway, 2004.

1.1.2.- ¿Qué es VoIP?

VoIP viene del inglés Voice Over Internet Protocol, es una tecnología que permite la transmisión de la voz a través de redes IP en forma de paquetes de datos.

Es un método que toma señales de audio analógicas (conversación telefónica convencional) y las transforma en datos digitales que pueden ser transmitidos a través de internet o de cualquier red de datos hacia diferentes direcciones IP.



Figura 1. 2.- VoIP

También se puede definir voz sobre IP como una aplicación de telefonía que puede ser habilitada a través de una red de datos de conmutación de paquetes vía protocolo IP (Internet Protocol).

Mientras la transmisión de datos e información ha sido hasta hoy en día la aplicación más prevaeciente en sistemas de información, el traslado de Voz sobre esta misma infraestructura ha generado grandes expectativas por el ahorro de recursos que ésta representa.

La tecnología telefónica utilizada en muchas regiones llamada Circuit-Switched implica la apertura de un canal digital permanente de punta-a-punta para establecerse la comunicación, mismo mecanismo que se traduce en un alto consumo de recursos, ya que una conversación no es llevada simultáneamente por ambas partes además de poseer una serie de silencios, ante estas deficiencias, nace la Voz sobre IP que tiene como su base la tecnología denominada Packet Switched en la que típicamente eran encapsulados datos para ser distribuidos a lo largo de un medio compartido.

El crecimiento y fuerte implantación de las redes IP, protocolos de transmisión en tiempo real, técnicas de digitalización de voz, así como el estudio de nuevos estándares que permiten que el servicio de voz sea más claro en cualquier red IP, han creado un entorno donde es posible transmitir telefonía sobre IP. Internet juega un papel crucial en esta tecnología. Es la posibilidad de estar comunicados a todo el mundo, es la puerta de entrada de nuevos servicios, a nuevas formas de comunicación apenas imaginadas.

Esta Tecnología consiste en la integración de datos y voz. Transporta las comunicaciones de voz por la web. En general, esto quiere decir enviar voz en forma digital en paquetes.

1.1.3.- ¿Cómo funciona?

La VoIP transforma la voz en "paquetes de información" manejables por una red IP (internet, intranets y extranet). Gracias a otros protocolos de comunicación, como el RSVP, es posible reservar cierto ancho de banda dentro de la red que garantice la calidad de la comunicación.

La voz puede ser obtenida desde un micrófono conectado a la placa de sonido de la PC, o bien desde un teléfono común: existen Gateway (dispositivos de interconexión) que permiten intercomunicar las redes de telefonía tradicional con las redes de datos.

La forma más común de VoIP es que funciona para el usuario final, establece una conexión de banda ancha de alta velocidad, un router y una puerta de enlace VoIP. En lugar de una línea telefónica normal, el router envía las llamadas telefónicas a través de una conexión a Internet. La puerta de salida VoIP, situado en algún lugar cerca de la conexión de Internet convierte la señal analógica en formato digital (ATA), que se desglosan en pequeños trozos llamados "paquetes", antes de enviarlo a través de Internet, como la forma en que se transmiten y desde el ordenador. Estos paquetes se envían a su destino final y las instrucciones para poner de nuevo en una forma comprensible, están incrustadas en ellos. A continuación, a través de una puerta de salida VoIP, donde los paquetes son reconvertida en el original formato analógico utilizando un PSTN (Red Telefónica Pública Conmutada).

1.1.4.- Estándares y Protocolos

1.1.4.1.-Protocolo H.323

Es una recomendación ITU que define los Sistemas de Comunicaciones Multimedia basados en paquetes y define una arquitectura distribuida para crear aplicaciones multimedia, incluyendo VoIP.

A finales de 1997 el VoIP fórum del IMTC debido a la ya existencia del estándar H.323 del ITU-T, que cubría la mayor parte de las necesidades para la integración de la voz, se decidió que el H.323 fuera la base del VoIP. A fin de evitar divergencias entre los estándares, se decidió que H.323 tendría prioridad sobre el VoIP. El VoIP tiene como

principal objetivo asegurar la interoperabilidad entre equipos de diferentes fabricantes, fijando aspectos tales como la supresión de silencios, codificación de la voz y direccionamiento, y estableciendo nuevos elementos para permitir la conectividad con la infraestructura telefónica tradicional. Estos elementos se refieren básicamente a los servicios de directorio y a la transmisión de señalización por tonos multifrecuencia (DTMF).

El VoIP/H.323 comprende a su vez una serie de estándares y se apoya en una serie de protocolos que cubren los distintos aspectos de la comunicación:

Característica	Descripción
Direccionamiento	<ol style="list-style-type: none"> 1. RAS (Registration, Admission and Status). Protocolo de comunicaciones que permite a una estación H.323 localizar otra estación H.323 a través del Gatekeeper. 2. DNS (Domain Name Service). Servicio de resolución de nombres en direcciones IP con el mismo fin que el protocolo RAS pero a través de un servidor DNS
Señalización	<ol style="list-style-type: none"> 1. Q.931 Señalización inicial de llamada 2. H.225 Control de llamada: señalización, registro y admisión, y paquetización / sincronización del stream (flujo) de voz 3. H.245 Protocolo de control para especificar mensajes de apertura y cierre de canales para streams de voz
Compresión de voz	<ol style="list-style-type: none"> 1. Requeridos: G.711 y G.723 2. Opcionales: G.728, G.729 y G.722
Transmisión de voz:	<ol style="list-style-type: none"> 1. UDP. La transmisión se realiza sobre paquetes UDP, pues aunque UDP no ofrece integridad en los datos, el aprovechamiento del ancho de banda es mayor que con TCP. 2. RTP (Real Time Protocol). Maneja los aspectos relativos a la temporización, marcando los paquetes UDP con la información necesaria para la correcta entrega de los mismos en recepción
Control de transmisión	<ol style="list-style-type: none"> 1. RTCP (Real Time Control Protocol). Se utiliza principalmente para detectar situaciones de congestión de la red.

Tabla 1. 1.- Características de H323

1.1.4.2.-Protocolo H.248 (Megaco)

También conocido como la IETF RFC 2885 fue desarrollado a partir de MGCP por lo cual tienen muchas similitudes. Es el estándar que permite que un media Gateway controller (MGC) controle a media gateways (MG). H.248 es el resultado de la cooperación entre la ITU y el IETF. H.248 se considera un protocolo complementario a H.323 y SIP.

Se utilizan cuando la función del Media Gateway es separada del Media Gateway Controller/Softswitch. Megaco es utilizado por el MGC para iniciación y control de media streams (como voz) a través del MG.

1.1.4.3.-AIX

“Inter Asterisk eXchange” es un protocolo abierto, es decir que se puede descargar y desarrollar libremente, este protocolo por ser nuevo no tiene mucho desarrollo más bien está en la fase de prueba para sistemas Asterisk por lo cual aun no es un estándar.

Es un protocolo de transporte, que utiliza el puerto UDP 4569 tanto para señalización de canal como para RTP (Protocolo de Transporte en tiempo Real).

Características	Puede truncar o empaquetar múltiples sesiones dentro de un flujo de datos, así requiere de menos ancho de banda y permite mayor número de canales entre terminales.
	En seguridad, permite la autenticación, pero no hay cifrado entre terminales.
	Puede usar cifrado (aes128), siempre sobre canales con autenticación MD5.

Tabla 1. 2.- Características de IAX.

1.1.4.4.-SIP

El protocolo Session Initiation Protocol (SIP) es un estándar de señalización que establece, encamina, modifica y termina llamadas o comunicaciones a través de redes IP, desarrollado por el IETF en 1999, el cual es un estándar que garantiza la universalidad de los sistemas de telefonía por Internet. También conocido como la IETF RFC 2543, define una arquitectura distribuida para crear aplicaciones multimedia, incluyendo VoIP.

Su sintaxis es similar a HTTP o SMTP, usa estas direcciones de tipo correo electrónico para identificar a los usuarios en vez de a los dispositivos. De esta forma SIP no depende del dispositivo y no hace distinción alguna entre voz y datos, teléfono u ordenador por tal motivo es el protocolo a usarse en este proyecto prototipo.

1.1.5.- ¿Por qué la Tecnología VoIP?

Se escogió esta Tecnología por su avance significativo y progresivo a lo largo de su desarrollo el cual ha sobrepasado a las compañías de telefonía convencional al grado que algunas de estas están migrando a esta tecnología, año con año este gran avance se ve reflejado en su cobertura por todo el mundo, así como también la convergencia entre tecnologías ya que no es necesario erradicar la telefonía convencional sino que pueden

fusionarse al grado de crear un sistema de comunicación más productivo y eficiente logrando las tan llamadas comunicaciones unificadas.

1.1.6.- ¿Donde se puede integrar esta Tecnología?

Esta Tecnología se puede emplear en cualquier lugar del mundo donde exista una conexión a internet o cualquier LAN(intranet, extranet ,etc.), cualquier red de datos es apta para implantar esta tecnología ya que VoIP ocupa esta misma para el envío voz por medio de paquetes atreves de la red de datos.



Figura 1. 3.- Tecnología VoIP.

1.1.7.- ¿Por qué en Luz y Fuerza?

El principal motivo es ahorrar, Luz y Fuerza es una institución la cual tiene un gran número de lugares o sectores los cuales abarcan la mayor parte del Distrito Federal y los cuales se encuentran conectados a la misma red de datos por diferentes medios fibra óptica, microondas y coaxial, los cuales viajan en las mismas líneas de luz aprovechando la gran



Figura 1. 4.- Logo LYF.

infraestructura que tiene la empresa por lo cual se garantiza el ancho de banda en cualquier lugar de la empresa el cual es de 100 mega bits.

Luz y Fuerza hoy en día gasta muchos recursos tanto monetario como horas/hombre en solucionar y dar mantenimiento a la red telefónica de la entidad. Por lo cual se pensó en una tecnología fácil de administrar e instalar para solucionar el problema así como también eliminar de tajo los grandes costos que la comunicación interna genera, para así solucionar un problema de la entidad.

1.2.- Alternativas

Existen muchas alternativas para tener una centralita VoIP entre las más comunes y que ofrecen mayor prestación están:

1.2.1.- Microsoft Office Communication Server 2007

En 2007 Microsoft OCS (Office Communications Server 2007) y Exchange Server 2007 juntaron sus fuerzas y así se unió VoIP. Exchange puede servir a Outlook o a Office Communicator (el cliente OCS) o sus equivalentes móviles. Exchange es sólo el objetivo, no la fuente. El correo de voz necesita conectividad PBX ahí es cuando entra OCS.



Figura 1. 5.- Office Communications.

1.2.2.- openPBX

OpenPBX es una rama privada del código abierto del sistema de Exchange (PBX) en el software para el sistema Linux. OpenPBX se encuentra bajo la licencia GNU (General Public License o GPL.)



Figura 1. 6.- Open PBX.

Este software fue desarrollado en PERL, contiene todas las prestaciones de una PBX, tiene un módico costo.

Entre sus características incluye su administración remota, integración con centrales telefónicas, interconexión entre diferentes OpenPBX.

OpenPBX tiene la particularidad de grabar sus conversaciones en sus líneas telefónicas y sistema de Mail to Fax / Fax to Mail (Recepción y envío de mails como fax).

1.2.3.- 3CX

Es un programa para Windows que probé de una centralita telefónica 3CX para Windows reemplaza completamente a los sistemas telefónicos propietarios tradicionales.



Figura 1. /.- 3CX.

Algunos de sus beneficios son:

Beneficios	No hay necesidad de cableado telefónico. Los teléfonos utilizan la red de computadores.
	Es más fácil de instalar y manejar, a través de la interfaz de configuración basada en web.
	Una IP PBX basada en software es mucho menos costosa que una PBX/PABX basada en hardware.
	Los empleados mudan oficinas sin hacer cambios en el cableado o configuración de la IP PBX.
	Elija entre varios teléfonos SIP basados en hardware sin quedar atado a un solo proveedor.
	Reciba y haga llamadas a través de la red PSTN estándar utilizando pasarelas VOIP.

Tabla 1. 3.- Beneficios 3CX.

1.2.4.-SIPX

Es una aplicación de código abierto, del protocolo SIP, basado en el sistema de comunicaciones IPBX. Permite conectar varios teléfonos para hacer llamadas entre sí, incluso permite la conexión a otros servicios del teléfono.

1.2.5.- Asterisk

Por último veremos Asterisk que es el software que se eligió para este proyecto por sus prestaciones que este ofrece, así como también el costo de este el cual es gratis con solo descargarlo de internet.

Usando el Concepto de Software libre Digium empresa emprendedora y desarrolladora primaria de Asterisk promueve el programa, el cual es un software para la creación de una centralita telefónica en solo una computadora o servidor. Asterisk puede funcionar en Linux y otras plataformas Unix aunque Linux sigue siendo las más soportada por este software.

Asterisk permite conectividad en tiempo real entre las redes PSTN y redes VoIP.
Sus características son:

características	Reducción de costos es extremadamente mínima si se trata de comunicación interna de la entidad.
	Tener control del sistema de telefonía.
	Ambiente de desarrollo fácil y rápido.
	Abundancia en recursos.
	Corre bajo Linux y es código abierto.

Tabla 1. 4.- Características de Asterisk.

CAPÍTULO 2

**ANÁLISIS DE FACTIBILIDAD PARA LA IMPLEMENTACIÓN DE TELEFONÍA
VoIP (ASTERISK) EN LYF.**

2.1.- Asterisk

2.1.1.- ¿Qué es Asterisk?

Asterisk es un software que permite la comunicación en tiempo real entre las redes PSTN (Public Service Telephony Network) y las redes IP. En otras palabras Asterisk transforma una computadora (PC o servidor) en una centralita telefónica (software) de código abierto.



Figura 2. 1.-Logo Asterisk

Asterisk es la solución más adecuada para este proyecto, es un software Open Source para la implementación de una IP-PBX (Conmutador IP), y es posible ejecutarla en la mayoría de los sistemas operativos aunque la principal plataforma y en la cual empezó es Linux se puede decir que es un software multiplataforma.

Asterisk fue originalmente escrito por Mark Spencer de DIGIUM, Inc. Después de él un gran número de programadores de varias partes del mundo se interesaron en la nueva tecnología y fueron la contribución de algunas fuentes abiertas de todo el mundo.

En la version 1.4.0 (Setiembre 2006), Asterisk ya contaba con:		
Líneas de código fuente		250.463
Estimación de número de desarrolladores en paralelo		25,08
Lenguajes de programación en Asterisk 1.4.0:		
Lenguaje	Líneas de código	Porcentaje
ANSI C	232.514	92,83%
sh	7.550	3,01%
c++	5.815	2,32%
perl	2.259	0,90%
yacc	1.508	0,60%
asm	642	0,26%
tcl	113	0,05%
PHP	62	0,02%

Tabla 2. 1.- Líneas de código en Asterisk.

Con la entrada de Asterisk más y más empresas van a poder experimentar recursos como URA (unidad de respuesta audible), DAC (distribución automática de llamadas), movilidad, buzón de voz, y conferencia, antes restrictivo solo a grandes compañías debido al alto costo.⁶

⁶ Flavio Goncalves, *Asterisk™ PBX, Guía de Configuración*, Rio de Janeiro, Edit. Independiente, 2007

La principal forma de comunicación en este tipo de tecnología es la VOIP que a su vez constituye una tecnología como tal, puesto que las redes de computo están en proceso de crecimiento y a su vez las comunicaciones unificadas están siendo cada día mas comunes en lo referente a las pequeñas y medianas empresas, ya que la implementación de este tipo de sistema necesita un hardware específico para conectar computadoras a la red, con que tenga una tarjeta de red es más que suficiente, lo que hace mucho más factible utilizarlo en empresas con una infraestructura suficiente como lo es Luz y Fuerza del centro. Asterisk un software que puede instalar en un PC o servidor y es capaz de emular una VoIP PBX a la cual se le pueden conectar dispositivos IP de cualquier tipo y que funcionen con casi cualquier protocolo de datos existente y sin restricciones. También es factible reutilizar los teléfonos antiguos con una simple interfaz (hardware) para realizar las tareas comunes de un teléfono tradicional y conectarlo a una PBX Asterisk.

En conclusión, el programa de licencia libre Asterisk me permite armar una PBX con una inversión casi nula en este proyecto ya que se ocupara la infraestructura de Luz y Fuerza del Centro (servidores, cableado estructurado, switch, teléfonos, PC, etc.).

2.1.2.-¿El por qué de Asterisk?

Uno de los primeros puntos y de los más importantes seria la reducción de costos en comparación de los precios que actualmente paga la compañía a la empresa que le provee de estos servicios. En este caso los costos se verían seriamente reducidos puesto que se eliminaran las extensiones de teléfono ya que estas serán sustituidas por nodos de red conjuntamente con una computadora auxiliada de un softphone o ya sea un teléfono normal acompañado de una interfaz ATA para poder conectarlo a la red de datos de la empresa la cual está presente en cada sector, campamento u oficinas centrales de Luz y Fuerza del Centro.

Tener el control total de la configuración de tu PBX en Software constituye un extra muy importante, puesto que Luz y Fuerza tendrá opción de modificar las opciones de las

cual provee Asterisk con lo cual quedara perfectamente constituido es sistema conforme a las necesidades actuales de la Empresa.

Características de Asterisk:

- Creación de extensiones.
- Envío de mensajes de voz a e-mail.
- Llamadas en conferencia.
- Menús de voz interactivos.
- Distribución automática de llamadas.
- En cuestión de Voip no se necesita ningún hardware adicional (Ethernet).
- Bloqueo de números no deseados por ID de llamada.
- Redireccionamiento de llamadas cuando la línea está ocupada.
- Bloqueo de no molestar.
- Música en transferencia.
- Identificador de llamadas.
- Re direccionamiento de llamadas cuando no hay respuesta.
- Video Conferencias.

2.1.3.- Antecedentes de Asterisk

Como ya antes se dijo Su creador es Mark Spencer (nacido en Alabama) creador de Asterisk, había creado en 1999 la empresa "Linux Support Services" con el objetivo de dar soporte a usuarios de Linux. Para ello necesitaba una pequeña central telefónica, pero ante la imposibilidad de adquirirla dados sus elevados costos, decidió construir una con una PC bajo Linux, utilizando lenguaje C.



Figura 2. 2.- Mark Spencer.

Posteriormente "Linux Support Services" se convertiría en el año 2002 en "Digium", redirigiendo sus objetivos al desarrollo y soporte de Asterisk.⁷

⁷ Historia de Asterisk, <http://bytecoders.homelinux.com/content/historia-de-asterisk-pbx.html>, [Consulta: Enero 2010]

2.1.4.- Ventajas

Una de las principales ventajas tiene que ver con la manutención de la red, esto quiere decir que es más barato darle mantenimiento a una red de datos global que darle mantenimiento a una red de datos y aparte a una red telefónica, esto se ve reflejado directamente en los bolsillos de la empresa Luz y Fuerza del Centro.

VoIP hace posible implementar una única red convergente que se encargue de transmitir todo tipo de comunicación, ya sea voz, datos, video o cualquier tipo de información.

El sistema implementado no requiere el establecimiento de un circuito físico durante el tiempo que toma la llamada, por lo que, los recursos que intervienen en la realización de una llamada pueden ser utilizados en otra cuando se produce un silencio, lo que implica un uso más eficiente de los mismos.

Las redes de conmutación por paquetes proveen alta calidad telefónica utilizando un ancho de banda menor que el de la telefonía clásica, ya que los algoritmos de compresión pueden reducir hasta 8kbps la digitalización de la voz produciendo un desmejoramiento en la calidad de la misma apenas perceptible.⁸

Tiene una extensa gama de recursos para proveer de características que hace apenas algunos años solo tenían las grandes empresas

El trabajo de mantenimiento de las redes telefónicas abarca muchos aspectos uno de los cuales era que si surgía algún problema se requería mandar a personal técnico a la empresa o al sector en que se requería o presentaba la falla, en el caso de Asterisk solo es cuestión de comunicarse por teléfono a la oficina encargada de la configuración del Asterisk y desde ahí o remotamente usted mismo podrá hacer los cambios pertinentes para que las extensiones funcionen como usted lo requiera.

⁸ Lógica básica de la voz sobre IP, <http://www.scribd.com/doc/14764672/VOIP>, [Consulta: Enero 20010].

A continuación listare las ventajas más comunes además de las citadas anteriormente:

- Reducción extrema de costos.
- Abundante en Recursos
- Es posible proveer contenido dinámico por teléfono.
- Plan de discado flexible y poderoso.
- Corre bajo software libre y es de código abierto.
- Gran adaptabilidad ante modificaciones
- Integración y convivencia con la telefonía tradicional
- Integración de voz y datos sobre una única red de comunicaciones
- Incremento de la productividad
- Ahorro en mantenimiento y gestión

2.1.5.- Sistemas Operativos soportados.

Originalmente Mark Spencer desarrollo Asterisk para el sistema operativo GNU/Linux, pero actualmente también se distribuye en varias versiones para los diferentes sistemas operativos BSD, MacOSX, Solaris y Microsoft Windows, aunque desde un inicio la plataforma nativa (GNU/Linux) es la mejor soportada de todas.

Un mejor funcionamiento de Asterisk se alcanza cuando no existen mas procesos que los que levanta el software Asterisk por si solo y sus anexos. Para que asi toda la potencia de la máquina y el procesador específicamente hablando se enfoque en los procesos que de Asterisk emanan.

Puesto que no necesitamos ningún entorno grafico para manejar Asterisk, la elección de Linux como Sistema Operativo es muy lógica ya que este sistema no necesita de un entorno grafico para hacer su función sin embargo se pueden instalar muchas alternativas de administración vía remota web para el mejor manejo y así no malgastar recursos para levantar un entorno grafico y todos los procesos subsecuentes de esta acción.

2.2.- Análisis de Factibilidad

2.2.1.- Comparación de costos.

Los costos son todos los desembolsos que debe hacer la empresa para implementar una nueva tecnología dentro de la misma.

TECNOLOGÍA	DESCRIPCIÓN
Red Multilínea (TELMEX) 	El costo por línea es de \$ \$178.03 mensual.
	Es el servicio complementario a Multilínea Negocio que le permite intercomunicar en forma ilimitada a sucursales, clientes o proveedores a través de un Plan Privado de Marcación, sin costo de Llamadas Locales ni de Larga Distancia Nacional sobre las líneas que contraten el servicio. ⁹
	Comunicación Ilimitada. Entre dos o más sitios Multilínea a través de su Plan Privado de Marcación. (Marcación a 4 dígitos)
ASTERISK (SOFTWARE LIBRE) 	El costo por línea es de \$00.00 mensual.
	Amplia gama de soluciones y características gratis.
	Comunicación Ilimitada. Entre dos o más sitios a través de la red de datos de la empresa (Marcación a 4 dígitos)
	Requiere servidor para su implementación.

Figura 2. 3.- Comparación de costos

2.2.2.- Reutilización e infraestructura

La infraestructura con que cuenta la empresa es basta en cuanto a red de datos se refiere ya que en cada sector se cuenta con un sistema completo de red local, por eso para este proyecto la red local juega un papel altamente importante, puesto que es la base de nuestro proyecto ya que por esta va a correr Asterisk.

⁹ Precios Telmex México, <http://www.telmex.com/mx/negocio/telefonía/local-red-multilinea.html>, [Consulta : Agosto 2010].

2.2.3.- Factibilidad de uso.

La investigación de factibilidad de este proyecto consta de saber el objetivo de esta tecnología, y lo que puede ofrecer a Luz y Fuerza del Centro, una vez visto esto se debe determinar si es útil que la tecnología sea implementada.

Una vez fijados los objetivos se deberá contemplar los recursos disponibles con que la empresa cuenta en este caso la red, computadoras, teléfonos, etc., son los recursos de la empresa actualmente y que serán reutilizados o explotados.

Al ocupar la red que ya existe podemos afirmar que más del 90 % de las llamadas que se realicen se obtendrá una muy buena calidad casi indistinguibles del servicio actual pero con mucho menos inversión, ya que cada llamada consume 17 Kbps de ancho de banda y la red de Luz y Fuerza tiene una capacidad de transmisión de 100 Mbps.

En el análisis costos/beneficios nuestro proyecto nos arroja cifras en las cuales el ahorro es el principal motivo por el cual en este nuevo sistema se dice que se ahorraría hasta el 80% en gastos de comunicaciones internas en Luz y Fuerza del Centro.

Debemos de tomar en cuenta la factibilidad motivacional que consta del análisis de la evaluación de las posibilidades de aceptación del nuevo sistema por parte de los administradores y usuarios en la cual para este proyecto ese tema de aceptación es muy bien recibido por parte de los usuarios y la administración es fácil y rápida.

Se necesita cubrir las metas con los recursos actuales en las siguientes áreas:

“LEGAL: El estudio de viabilidad legal debe informar si la legalidad vigente permite, o más bien no impide la realización del proyecto. Este estudio de viabilidad, en lo posible debe ser efectuado por expertos y el encargado de evaluar la factibilidad global del proyecto ha de seguir la orientación de los especialistas.”¹⁰

En este proyecto sería de gran importancia la viabilidad legal ya que al ser una empresa gubernamental se requiere por fuerza de licencias de uso del software, aunque en este caso el software Asterisk está distribuido bajo la Licencia de Software libre lo cual no impide la implantación de esta tecnología en Luz y Fuerza.

¹⁰ Factibilidad de uso, http://www.fonaes.gob.mx/pdf/capacitacion/estudio_de_factibilidad.pdf, [Consulta: Enero 2010].

“TÉCNICA: Este estudio determina si la tecnología disponible, permite hacer realidad el proyecto y también si es conveniente hacerlo. Un estudio de viabilidad técnica debe proveer información sobre las diversas formas de materializar el proyecto o los diferentes procesos que pueden utilizarse para producir un bien o servicio. El estudio deberá contener una estimación de los requerimientos de capital, mano de obra y recursos materiales, tanto para la puesta en marcha, como para el estado de operación del proyecto.”

La tecnología propia de Luz y Fuerza comprende una gran variedad, que como ya anteriormente se comentó de piezas que juegan un rol muy importante junto con la tecnología a implantar, con estos datos podremos decir que conviene la implantación de esta tecnología, puesto que conlleva a reutilizar los recursos con los que cuenta la empresa y así con la inversión mínima para la adaptación de algunos componentes.

Decimos que el proyecto es rentable por muchos aspectos referentes a:

1. Ahorro en equipos de conmutación.
2. Ahorro en licenciamiento de sistemas para su operación.
3. Ahorro en mantenimiento de red.
4. Ahorro por pago de líneas a Telmex dentro de LYF.
5. Ahorro en reutilización de material ya existente en la entidad.
6. Ahorro en Instalación y Puesta en Marcha (Configuración del sistema, actualización, mantenimiento serán realizadas por Personal de la misma empresa).
7. Ahorro en conexiones, cableado, acondicionamiento de oficinas de modo que se reutilizaran los nodos de red o las computadoras existentes.
8. Ahorro en operación del sistema, en respaldos y contingencias, etc. ya que el personal encargado de ese trabajo será de Luz y Fuerza.
9. Ahorro en mantenimiento especial del hardware y software puesto que será como un servidor.
10. Ahorro en capacitación a usuarios, administrativos y personal técnico ya que en vez de contratar una empresa para esto se podrá hacer dentro de la empresa con ayuda del mismo personal de la empresa.

2.2.4.- *Licencias*

El derecho de autor o copyright sirve para decir que tal obra tiene su dueño de manera que toda persona que desee hacer uso de la obra de otro necesitará obtener la autorización de éste.¹¹

Las licencias de software libre son un mecanismo legal en el cual en este caso permite que el software sea adaptado a las necesidades del usuario.

Un Software sin acceso al código fuente o código de programación tiene restricciones legales y técnicas lo que causa que el usuario del software tenga limitaciones para mejorarlo, traducirlo, adaptarlo a sus requerimientos o a los de terceros, distribuirlo y hacer uso de él.

Frente a tal limitación y en contraparte surge un movimiento llamado “free software” que desde el punto de vista de los programadores significa tener ciertas libertades en cuanto al código fuente:

1. La libertad de usar el programa, con cualquier propósito
2. La libertad de estudiar cómo funciona el programa
3. La libertad para adaptarlo o modificarlo acorde con las necesidades del usuario
4. La libertad de distribuir copias del programa, sea modificado o no.

“Open source” o “software de código abierto” significa que el desarrollador desde un principio da acceso al código fuente, con el fin de que el usuario pueda modificarlo, ósea que el creador concede desde antes de distribuirlo, el derecho a hacer, acceder, usar, modificar y distribuir el programa.

- » GNU Public License (GPL) (creada a mediados de los 80)
- » Lesser GNU Public License (LGPL)



Figura 2. 4.- GNU

¹¹ Licencias de Software, <http://www.agendadigital.cl/files/UsodeSoftwareLibreEstado.pdf>, [Consulta : Enero 2010]

- » Artistic License
- » Berkeley Software Distribution (BSD)
- » Massachusetts Institute of Technology License (MIT)
- » Sun Public License (SPL)
- » Mozilla Public License (MPL)
- » European Union Public License (EUPL)

Asterisk fue liberado bajo la Licencia Pública General de GNU (*GNU General Public License-GPL*). Free Software Foundation como ya antes se menciono creó esta licencia a mediados de los 80, y está orientada principalmente a proteger la libre distribución, modificación y uso de software. Su propósito es declarar que el software cubierto por esta licencia es software libre y protegerlo de intentos de apropiación que restrinjan esas libertades a los usuarios.

La licencia GPL, al ser un documento que cede ciertos derechos al usuario, asume la forma de un contrato, por lo que usualmente se la denomina contrato de licencia o acuerdo de licencia.

Licencia GPL2 (General Public License):

1. Permite ejecutar el programa con cualquier propósito.
2. Permite modificar el programa (acceso al código fuente).
3. Permite redistribuir copias.
4. Permite mejorar el programa y publicar estas mejoras (junto al código fuente).
5. Los añadidos de terceros al código de Asterisk implican la cesión de los derechos comerciales a Digium pero se conserva el copyright del desarrollador. El nuevo conjunto se libera como GPL2.

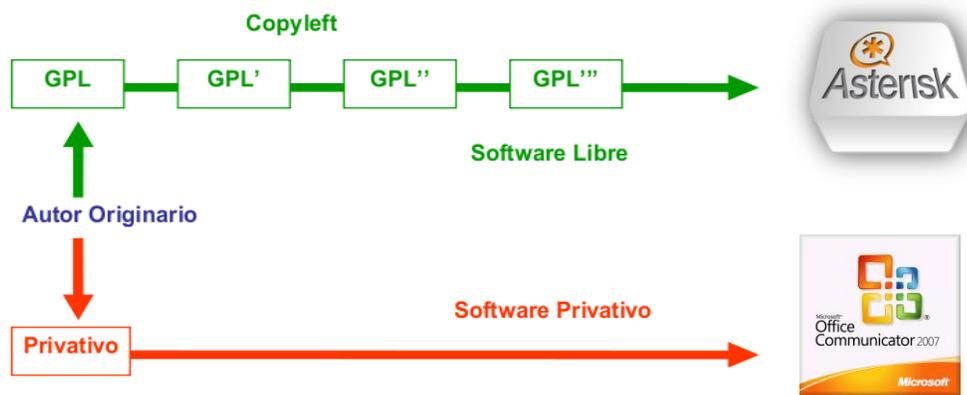


Figura 2. 5.- Licencia GPL2.

Comparativa de costos:	
Office Communications Server 2007 Enterprise Edition	\$3.999 Dólares
Asterisk solución libre	\$0.00 pesos

Tabla 2. 2.- Comparativa de Costos de Software.

2.2.5.- Capacitación de usuarios finales

Hoy en día las comunicaciones, dentro de una organización, constituyen una parte crucial para el desarrollo de las distintas actividades empresariales, y representan las bases que permiten conducir con éxito a las organizaciones en nuestro caso Luz y Fuerza en mercados altamente competitivos.

Para lo anterior la capacitación juega un papel esencial en la introducción de nueva tecnología de comunicación, más aun, como en el caso de Luz y Fuerza del Centro, cuando incluye implícitamente nuevas formas de pensamiento (técnicamente hablando) para la entidad.

En el caso de la capacitación en materia del nuevo sistema de comunicación para la entidad, es necesario identificar las diferentes categorías del personal a capacitar: personal directivo, personal técnico y usuarios finales. A continuación se describen los principales aspectos a tener en cuenta para cada una de las categorías anteriores:

- **Directivos:** Para los directivos debe realizarse exhaustiva capacitación para enseñarles los aspectos más básicos pero también los alcances de esta tecnología para que sepan aunque no lo realicen ellos mismos como tal las características de

nuestro sistema de comunicación interna, para que tengan en su conocimiento lo que se puede alcanzar y así sacar más provecho de esta tecnología.

- **Personal técnico:** Para el personal técnico se realizarán cursos introductorios así como capacitación constante para estar al día con diferentes actualizaciones del software y se dividirán en 3 grupos: Personal de administración de servicios, personal de soporte técnico al usuario final, y personal de desarrollo.
- **Usuarios finales:** Para la capacitación de usuarios finales, se deberá tomar en cuenta algunos aspectos que en mi experiencia en la empresa y me atrevo a decir que en muchas empresas más la principal barrera es sin lugar a dudas el miedo a lo desconocido, para estos usuarios será necesario la capacitación de una manera mas sencilla pero sin dejar a un lado los alcances, así a los usuarios menos experimentados en esta tecnología se complique menos su captación.

Por lo anterior podemos decir que la migración a este tipo de tecnología se puede asimilar muy rápidamente y con esto la funcionalidad de la empresa no se verá seriamente afectada por causa de la aparición de nuevos procedimientos de comunicación.

2.2.6.- Conclusión de la factibilidad

Como conclusión y gracias al estudio de las diferentes factibilidades aquí presentadas, se puede decir que el sistema que se piensa implementar es una manera más eficaz de comunicar a cada una de las áreas dentro de la red de luz y fuerza del centro, ya que al momento de emplearla se verá reflejado directamente en la amplitud de posibilidades que esta tecnología lleva consigo, así como en la economía de la misma ya que al tener un gasto inferior en lo ya antes se menciona tendrá solvencia para áreas donde más lo necesiten, y así mejorar la economía de la empresa.

CAPÍTULO 3

INTRODUCCIÓN A LA TECNOLOGÍA VOIP

3.1.- Introducción

La Voz sobre IP (VoIP, Voice over IP) es una tecnología que permite la transmisión de la voz a través de redes IP en forma de paquetes de datos.

Desde la invención del teléfono por Alexander Graham Bell a finales del siglo XIX, pareciera que las formas de comunicación no han cambiado brutalmente. Apenas hace algunas décadas la noticia eran los teléfonos móviles o celulares, sin embargo las comunicaciones de empresas y oficinas así como las comunicaciones de nuestras casas permanecieron sin cambios drásticos.

La definición de muchos especialistas en el ramo para esta tecnología sería: El proceso de fragmentar el audio y el video en paquetes pequeños, transmitirlos por la red en cuestión y desfragmentarlos durante la recepción de tal manera que dos personas puedan comunicarse.¹²

La VoIP es una tecnología que está creciendo de una manera muy rápida, inclusive las empresas proveedoras de acceso a internet por ejemplo Telmex en México están migrando a sus redes tradicionales que estaban basadas en conmutación de circuitos a redes que soporten VoIP para ofrecer mejor calidad, nuevos servicios y sobre todo a menor costo.

Pero este no fue sino hasta saber el gran alcance que esta nueva tecnología proporcionaba, ya que en principio no querían implementarla tanto que Telmex en 2005 creó una cláusula (cláusula 18) en su contrato la cual prohíbe a sus clientes cursar voz sobre IP a través de Prodigy Infinitum.

Esta última acción de parte de la empresa con más alto número de usuarios de internet en México dejó de ser válida dos años después en abril del 2007 con la noticia de que Telmex cambiaba su cláusula décimo octava para así permitir llamadas vía internet y hacer que los clientes pudieran utilizar el servicio como un medio de transmisión de voz sobre IP.

Lo que básicamente dice la cláusula es que el usuario al contratar en servicio de acceso a internet de banda ancha infinitum pueda acceder a cualquier servicio, aplicación o contenido que se encuentre disponible en el internet.

Toda la historia que aunque es poca, resulta muy clara y concisa y sobre todo apunta hacia la migración de la telefonía tradicional a la telefonía IP. Así pues, en una opinión

¹² Telefonía IP, <http://www.lsidotcom.com/productos/telefonía/intro>, [Consulta : Enero 2010]

particular, en un futuro próximo, toda la comunicación que conocemos ya sea voz, datos y video se haga por medio de la tecnología VoIP.

VoIP nos sirve para enviar señales digitales de la voz, igualando la calidad de del sonido con la telefonía tradicional inclusive superándola, todo esto en un tiempo real. Una de las preguntas que muchos de los que no están familiarizados con esta tecnología sería ¿Qué podemos hacer con VoIP? Pues una cosa muy sencilla para una persona normal hablar con quién queramos y cuando queramos así este en el otro lado del mundo o en este caso en cualquier lugar de Luz y Fuerza del Centro.

Un Software muy Popular en Internet se llama Skype, el cual permite realizar llamadas vía Internet a larga distancia con la tecnología punto a punto (PC a PC) se puede comparar al tan famoso Messenger de Microsoft pero en vez de mensajes se trata de voz. Funciona como un medio de comunicación de voz gratuito, el inconveniente sería que es necesario poseer el mismo programa para así poder comunicarse.

3.1.1.-Telefonía Tradicional

3.1.1.1.-Historia

Todo empezó con la creación del Teléfono y lo inicio el científico Alexander Graham Bell (1847-1922) debido a que su esposa y madre eran sordas se dedicó a experimentar con aparatos para los oídos combinados con el telégrafo hasta lograr



Figura 3.1.- Telégrafo

desarrollar el *Teléfono* al cual lo patento en América en el año de 1876.



Figura 3.2.- Alexander Graham Bell inaugurando en octubre de 1892 el servicio telefónico

El 10 de marzo de 1876, Bell y Watson (su ayudante) logran transmitir la primera la señal de voz a través de un cable eléctrico, la cual se convirtió en la frase más famosa dentro de la historia de las Telecomunicaciones y fue: “Mr. Watson, come here, I want you!” (“Sr. Watson, venga aquí, lo necesito!”).

Hasta hace poco se demostró que el Italiano Antonio Meucci (1808-1889) es el verdadero inventor del teléfono, así lo reconoció el Congreso de los Estados Unidos. Meucci era inmigrante muy pobre y no fue sino hasta 1871 cuando obtuvo la patente en Nueva York sin embargo, en 1873 debía renovar la patente, cosa que no hizo por carecer de 10 dólares.

Así pues en el 2002, el congreso de Estados Unidos reconoció, casi 130 años más tarde que el padre del Teléfono fue un Italiano llamado (Meucci), considerado un gran hombre por haber abierto el camino a la comunicación vía cables.



Figura 3. 3.- Antonio Meucci.

Después de la controversia por la invención del Teléfono, En el año de 1877 en New Haven (Connecticut, EE.UU.) empezó a funcionar la primera central telefónica, siguiéndole los países de Francia e Inglaterra con centrales telefónicas similares.

En un comienzo la primera central funcionaba prestando el servicio de 21 abonados y sólo tenía 8 líneas telefónicas individuales. Para los años de 1880, solo tres años más tarde ya prestaban servicios más de 408 centrales telefónicas distribuidas por todo el territorio norteamericano. Hablando de la telefonía en México la primera central telefónica empezó a funcionar en 1878 estableciendo un enlace entre la Ciudad de México y en aquel tiempo el pueblo de Tlalpan, fue un enlace de aproximadamente 16 km, meses más tarde se anuncia la primera red de telefonía cuya función era la de conectar a las oficinas de la Secretaria de Guerra, durante el mandato de Porfirio Díaz.

Almon Brown Strowger (1839-1902) en el año de 1892 da a conocer su Central Telefónica Automática motivado por su negocio, dado que él era dueño de la empresa cuya función era dar servicios funerarios en la región de Kansas a quien se dirigían todas las

personas que buscaban este servicio, hasta que por fin tuvo competencia y miro que ya no tenía un monopolio y que ahora la competencia le hacía perder ritmo al negocio. Dado que quien tenía mejor relación con las operadoras de la central telefónica más cercana se llevaba a todos los clientes. Motivado por esta circunstancia se dio a la tarea de solucionar el problema de infinidad de negocios que estaban siendo llevados a la quiebra por la corrupción y diseño el primer Sistema Electromagnético de Conmutación Telefónica el cual hacía que la mediación de las operadoras fueran desechadas las llamadas telefónicas, siendo este el cambio más importante desde la creación del teléfono.

La primera Central Telefónica Automática del mundo fue instalada en Indiana (EEUU), en 1893. Y a los pocos años los diferentes paises hicieron lo mismo.

Con la implementación de las centrales telefónicas automáticas la espera de conexión de acorto de 15 minutos que se tardaba una central manual a aproximadamente 3 minutos con una central automática. Así como también se garantiza la confidencialidad de las llamadas.

Para tener una mejor comunicación y a más distancia desde que existía el telégrafo, hubo “cables marinos” que en principio era para conectar islas o países que estaban a poca distancia, separados por algún estrecho y un poco de mar. Desde 1866 estos cables existen haciendo posible la comunicación vía telégrafo, uniendo a finales del XIX a Gran Bretaña, Francia, Alemania, Estado Unidos y Canadá.

En 1895 el ruso Aleksandr Stepánovich Popov (1859-1905) encontró el mejor sistema para enviar y captar las ondas electromagnéticas, construyendo la antena con hilo metálico. Poco tiempo después, en el año de 1895 Guglielmo Marconi (1874-1937) a ese invento le añadió un receptor de hilo en sentido vertical (Antena) para poder captar mejor las oscilaciones eléctricas. Las Autoridades Italianas no mostraron mucho interés así que Marconi decidió trasladarse al Reino Unido, cuyo apoyo lo recibió del jefe de Correos, haciéndole a su invento modificaciones para poderlo patentar en el año de 1896. Nuevamente hubo una controversia de quien fue el



Figura 3.4.- Guglielmo Marconi

inventor de la radio telefonía, Popov presento casi al mismo tiempo un receptor de ondas similar al de Marconi. Creo una asociación con su primo Jameson Davis, y fundaron la compañía Wireless Telegraph and Signal Company, Ltd.

En el año de 1956 se tiende el primer cable transatlántico de la historia capaz de mantener 36 líneas telefónicas al mismo tiempo. Hasta la fecha estos cables se siguen utilizando puesto que en 1988 se tendió fibra óptica la que se extiende por debajo del atlántico, siendo capaz de realizar 40,000 llamadas al mismo tiempo, quedando la red como lo muestra la imagen siguiente.

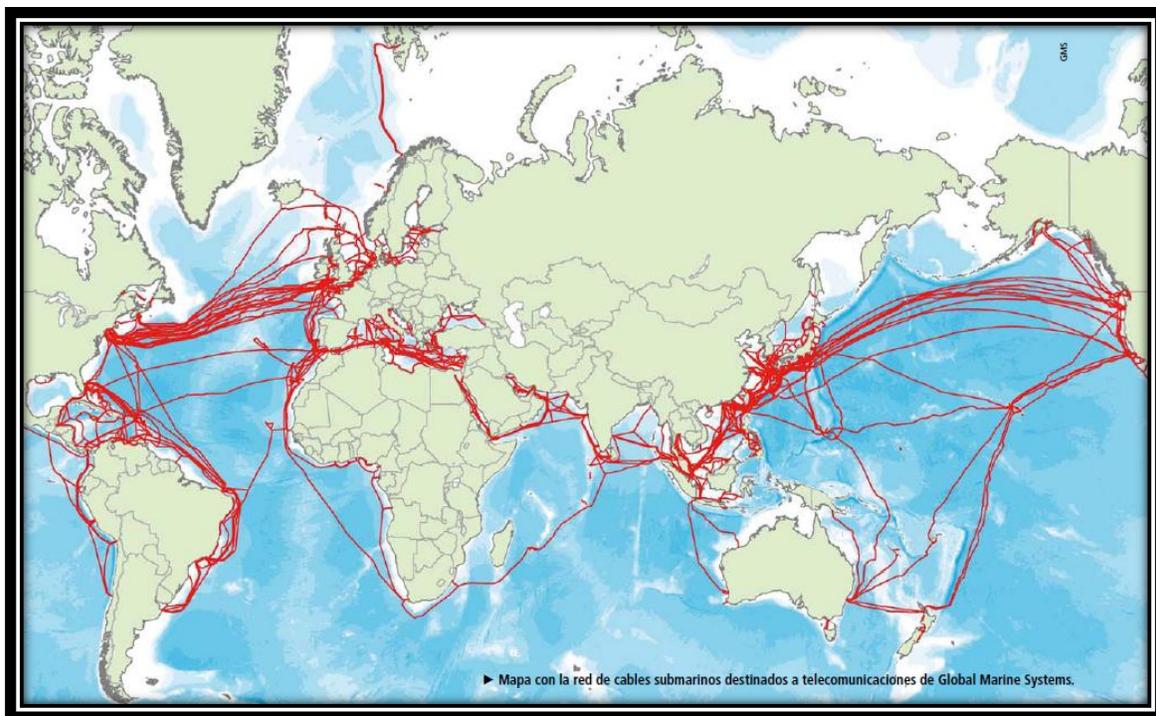


Figura 3. 5.- Mapa del cableado marítimo

Más adelante se logra poner en órbita el primer satélite de comunicaciones el Telstar one en el año de 1962, gracias a la financiación comercial de varios consorcios estuvo a cargo de AT&T colaborando con los laboratorios Bell. Fue diseñado para transmitir la señal de televisión en directo y las conversaciones telefónicas intercontinentales. En este satélite se realizó la primer llamada satelital de la historia así como la primera transmisión de fax, datos y televisión en directo vía satélite.

Las computadoras fueron desde sus inicios una herramienta eficiente y de grandes proyecciones, y uno de los primeros en la investigación de comunicaciones entre computadoras fue Advanced Research Projects Agency (ARPA), quienes lograron el primer enlace entre computadoras en 1965 a través de un canal dedicado de 1200 bits por segundo. Para el año de 1969 se logra la primera red de computadoras de la historia llamada ARPANET, y estaba compuesta por cuatro nodos:

1. UCLA (Universidad de California en los Angeles)
2. SRI (Stanford Research Institute)
3. UCBS (Universidad de California de Santa Bárbara, Los Ángeles)
4. Universidad de UTA.

ARPANET en los 70's es el predecesor del protocolo Host-to-host llamado NCP el cual es el predecesor del actual TCP/IP no mucho tiempo después crean ALOHANET que fue la primera red de conmutación de paquetes vía radio y para 1972 estas dos redes se unirían para así lograr una red mucho más grande y convergente entre tecnologías, el año siguiente se crea la primera conexión internacional de la ARPANET la cual incluye al colegio universitario de Londres.

En los años de 1973-1974 se crea TCP (*Transmission-Control-Protocol*) el cual es un protocolo fundamental de la Internet. Este protocolo garantiza los datos, ya que confirma que sean entregados los paquetes a su destino y sin errores, también proporciona la cualidad de distinguir aplicaciones dentro de una misma máquina, a través del concepto de puerto. Al mismo tiempo aparece el Protocolo IP (*Internet Protocol*) desarrollado por el mismo informático que diseñó el TCP llamado Vinton Cerf para conectar varias redes de distintas universidades, ya para el año de 1982 nombran al protocolo TCP y al protocolo IP como el conjunto de protocolos TCP/IP de comunicación a través de la ARPANET.

Paralelamente se creó la técnica de conmutación de paquetes, la cual consiste en transmitir los datos en pequeños paquetes, estos pequeños paquetes son adicionados de una serie de bits de control. La ventaja de esta técnica es que los paquetes son enviados por varias rutas haciendo la transmisión en paralelo, además si por cualquier motivo ocurre algún error solamente se retransmite el paquete con el error y no todo el mensaje.

Esta técnica esta creo dos métodos los Datagramas y los Circuitos Virtuales, en el método de Datagramas no todos los paquetes siguen una misma ruta, tampoco un orden debido a que el tratamiento de los paquetes es independiente.

Por otro lado en los circuitos virtuales la transmisión se establece previa a la transmisión de los paquetes por medio de paquetes de Petición de Llamada, una vez establecida la llamada empieza el intercambio de datos los paquetes se recibirán en el mismo orden en que fueron enviados.

La primera red que uso conmutación de paquetes fue española y se llamaba IBERPAC.

Durante el año de 1972, Stromberg-Carlson (fabricante de equipos de telecomunicaciones) fue quien desarrollo la primera central telefónica digital (la “DBX”), debido a que el avance de la tecnología estas centrales lograron tener una reducción brutal de su tamaño y consumo eléctrico, además de que regulasen todos los movimientos de una central telefónica convencional.

Justo es ese año otro avance dentro de las telecomunicaciones surgía en un estudiante de doctorado del MIT (*Massachusetts Institute of Technology*) Robert Metcalfe cuya tesis constaba de desarrollar un protocolo más eficiente que los que existían en esos momentos y desarrollo el CSMA/CD (Carrier Sense Multiple Access / Collision Detection). Ya con 27 años de edad y trabajando para el Centro de Investigaciones de Xerox en Palo Alto se le encomendó la tarea de diseñar y construir la red que uniera todo aquello, con la ayuda de su colaborador David Boggs experimentaron con varios prototipos de red hasta que construyeron “Alto Aloha Network”. Este prototipo lo fueron mejorando hasta que el 22 de mayo de 1973 lo que hoy conocemos como Ethernet la cual tenía todas las características básicas de la Ethernet actual. Empleaba CSMA/CD para minimizar la probabilidad de colisión, y en caso de que ésta se produjera se ponía en marcha un mecanismo denominado retroceso exponencial binario para reducir gradualmente la ‘agresividad’ del emisor, con lo que éste se adaptaba a situaciones de muy diverso nivel de tráfico. Tenía topología de bus y funcionaba a 2,94 Mb/s sobre un segmento de cable coaxial de 1,6 km de longitud. Las direcciones eran de 8 bits y el CRC de las tramas de 16 bits. El protocolo utilizado al nivel de red era el PUP (Parc Universal Packet) que luego

evolucionaría hasta convertirse en el que luego fue XNS (Xerox Network System), antecesor a su vez de IPX (NetWare de Novell).¹³

El Dr. Martin Cooper es considerado el inventor del primer teléfono portátil. Considerado como "el padre de la telefonía celular"; siendo gerente general de sistemas de Motorola realizó una llamada a sus competidores de AT&T desde su teléfono celular, transformándose en la primera persona en hacerlo.



Figura 3. 6.- Dr.Martin Cooper.

Comenzando la década de los 80's apareció la Red Digital de Servicios Integrados (RDSI o ISDN) el cual es un protocolo estándar de red de comunicaciones, que contempla tanto las comunicaciones de voz, como las de datos, transmitiendo ambas en formato digital, y a distintas velocidades, todas ellas más rápidas y seguras que la línea analógica convencional de teléfono RTB (Red Telefónica Básica).

El 26 de febrero de 1991, Tim Berners Lee presentaba WorldWideWeb, un programa para la estación de trabajo Next que implementaba el primer navegador web del mundo. Y con esto dar paso al concepto de Internet, no tardaría mucho para que actualizaran estas tecnologías y crear internet como lo conocemos actualmente. Actualmente, Tim Berners Lee es presidente del World Wide Web Consortium, un organismo creado para dirigir el desarrollo de las tecnologías relacionadas con la web.

Este es el medio por donde se transmitirá nuestro sistema de comunicación interna en Luz y Fuerza, el sistema en primera etapa no estará conectado directamente a internet pero si ocupara los principios básicos de este.

¹³ Comunicaciones, <http://es.wikipedia.org/wiki/Ethernet>, [Consulta: Febrero 2010]

3.1.1.2.-PSTN

PSTN significa red telefónica pública conmutada. La cual es una red analógica.

La red de telefonía pública conmutada (PSTN, *Public Switched Telephone Network*) es una red con conmutación de circuitos tradicional optimizada para comunicaciones de voz en tiempo real¹⁴. Otra definición pudiera ser la concentración de las redes públicas mundiales de circuitos conmutados para establecer comunicación entre dos terminales.

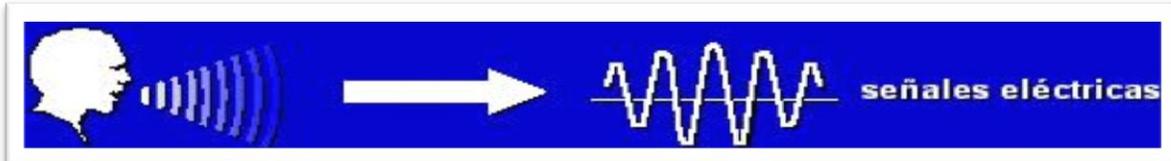


Figura 3. 7.-PSTN (transporta señales eléctricas).

3.1.1.3.-RDSI o ISDN

La red digital de servicios integrados (RDSI o ISDN) es la evolución de la PSTN, lo que hace es hacer una conexión entre dos o más extremos en un nivel digital para así tener un mejor servicio y gozar de más prestaciones. Se dice servicios integrados porque utiliza la misma infraestructura para muchos servicios que tradicionalmente requerían interfaces distintas, la transmisión es digital ofreciendo una capacidad básica de comunicación de 64 Kbps.



Figura 3. 8.-ISDN (Integrated Services Digital Network)

¹⁴ Conmutación de Circuitos, http://www.microsoft.com/windows/windows2000/es/advanced/help/sag_TAPIconcepts_116.htm, [Consulta : Febrero 2010]

3.1.2.-Telefonía IP

La Telefonía IP es una rama de la tecnología RDSI, de forma que se pueden establecer llamadas telefónicas sobre redes IP simplemente utilizando un PC, gateways y teléfonos estándares.

Hace varios años ya se pensaba en la convergencia de las redes, en otras palabras unir datos, voz y video en una sola red sin tener que estar separadas. Este

tema formaba parte del pensamiento de los grandes empresarios, hasta que hace poco esos pensamientos se transformaron en realidad , por ejemplo Microsoft ya está empleando la tecnología de VoIP en su Windows Live Messenger, así como esta empresa hay infinidad de empresas en busca de las ventajas que te da esta tecnología.

En un inicio esta tecnología era muy costosa pero con el paso del tiempo se fueron abaratando los equipos terminales así como soluciones básicas como los softphone, todo esto ha influido para que las empresas incursionaran en esta índole de la VoIP.

También la calidad de la voz ha ido mejorando con el paso del tiempo, hoy en día es imperceptible llamar por una red PSTN o por VoIP.

3.1.2.1.-Origenes.

Aquí en Latinoamérica el concepto de Telefonía IP se inició hace algunos años con las primeras implementaciones de telefonía IP. El primer caso del que se tiene conocimiento en Colombia es el de la Universidad San Buenaventura, en Bogotá, que hacia 1999 instaló una red convergente (de voz y datos), con tecnología de 3Com. La Telefonía IP conjuga dos mundos históricamente separados: la transmisión de voz y la de datos. Se trata de transportar la voz, previamente convertida a datos, entre dos puntos distintos.

El origen de Voip y de los softphone es Skype, el cual se remonta a los tiempos en que sus fundadores imaginaban que algún día existiría algo para que la gente hablara gratis



Figura 3. 9.-Conexión de teléfono.

a distancia¹⁵. Durante su creación, se referían a su producto como Sky peer-to-peer (cielo de igual a igual), aludiendo a que su aplicación sería como el cielo, sin fronteras, y usaban la abreviatura Skyper. Pero se encontraron con que Skyper era un dominio ya registrado, así que decidieron dejarlo en Skype. Suena bien y, aunque no significa nada, deja volar la imaginación. Skype es hoy una de las marcas más reconocidas en el mundo, además de que es una empresa impulsadora de la Telefonía IP. Las llamadas entre usuarios de Skype son gratuitas, y llamadas a teléfonos externos a la red disfrutan de unas tarifas muy reducidas que junto con el aumento de la banda ancha a nivel mundial hacen de la telefonía IP una de las opciones más idóneas para este tipo de llamadas.

Esto posibilitaría utilizar las redes de datos para efectuar las llamadas telefónicas, y observando un poco más allá, desarrollar una única red que se encargue de cursar todo tipo de comunicación, ya sea vocal o de datos. Es evidente que el hecho de tener una red en vez de dos, es beneficioso para cualquier operador que ofrezca ambos servicios.

En Agosto de 2005, Google lanzó Google Talk, un software de mensajería instantánea y servicio de voz IP que compite directamente con Skype. Su principal innovación frente a Skype es que usa un protocolo abierto para la comunicación, lo que permite a los usuarios crear mejoras para el programa.

Sin embargo, a día de hoy, este software cuenta con el gran inconveniente de que sólo se puede ejecutar desde Windows 2000/Xp/2003, lo que limita su utilización entre los usuarios de Computadores. En Septiembre del 2005, Microsoft anunció la compra de Teleo, una pequeña empresa de San Francisco que dedicaba su negocio a la telefonía por Internet. Es un hecho que la Telefonía ha evolucionado, a tal grado que ya no estamos hablando de características telefónicas, sino más bien de soluciones de negocio en comunicaciones, en Luz y Fuerza el teléfono ha pasado de ser un elemento más de la red de la empresas a una fuente de información o generador de ingresos. Además al adquirir inteligencia con VoIP, se puede otorgar facilidad, adaptabilidad y flexibilidad en el sistema interno de la entidad, y aumentar su productividad y satisfacción de sus clientes.

Hace algunos años se venía manejado un máximo de 256 kbps en las compañías proveedores de internet, ahora la velocidad promedio de una conexión a internet es de 2048

¹⁵ Skype, <http://www.oobo.com/origen-nombre-skype.html>, [Consulta : Febrero 2010]

kbps entrante y 384 kbps saliente, lo cual no genera problemas de saturación de banda en una llamada Voip.

Otro dato importante es sin duda el nivel de seguridad que ofrecen estas redes, ya que la seguridad es infinitamente superior, pues la empresa de Luz y Fuerza no tienen que preocuparse de dos redes diferentes, y tratar sólo con una red en la que convergen los datos, la voz y el vídeo.

3.1.2.2.- ¿Cómo funciona la Telefonía IP?

Los primeros o los pasos básicos que tienen lugar en una llamada a través de Internet son: conversión de la señal de voz analógica a formato digital y compresión de la señal a protocolo de Internet (IP) para su transmisión. Del otro lado en la recepción se realiza el proceso inverso para poder recuperar de nuevo la señal de voz analógica.

Cuando hacemos una llamada telefónica por IP, nuestra voz se digitaliza, se comprime y se envía en paquetes de datos IP. Estos paquetes se envían a través de Internet a la persona con la que estamos hablando. Cuando alcanzan su destino, son ensamblados de nuevo, descomprimidos y convertidos en la señal de voz original.

Hay tres tipos de llamadas:

- PC a PC, siempre gratis.
- PC a Teléfono, gratis en algunas ocasiones, depende del destino.
- Teléfono a Teléfono, muy baratas.

Como ya se dijo antes en esta tesis, en una llamada telefónica normal, la PBX establece una conexión permanente entre ambos extremos, conexión que se utiliza para llevar las señales de voz. En una llamada telefónica por IP, los paquetes de datos, que contienen la señal de voz digitalizada y comprimida, se envían a través de Internet a la dirección IP del destinatario. Cada paquete puede utilizar un camino diferente para llegar, están compartiendo un medio, una red de datos. Cuando llegan a su destino son ordenados y convertidos de nuevo en señal de voz.

Lo revolucionario de la idea de VoIP es el no utilizar circuitos físicos para las conversaciones, sino que se envían múltiples conversaciones a través de un mismo canal, de manera que se hace un uso mucho más eficiente de los recursos. Los algoritmos empleados son capaces, por ejemplo, de aprovechar los tiempos de silencio dentro de una conversación para transmitir datos correspondientes a otras. Son evidentes las ventajas de una red inteligente de este tipo, ya que la misma infraestructura puede prestar más servicios de una mejor calidad.

Básicamente, el proceso comienza con la señal analógica original que es digitalizada en señales PCM (Pulse Code Modulation) por medio del codificador/decodificador de voz (códec). Estas muestras PCM (que no son más que una lista de ceros y unos) son pasadas al algoritmo de compresión, el cual procesa la voz y la divide en paquetes que pueden ser transmitidos a través de una red. En el otro extremo de la conexión se realizan exactamente las mismas funciones en un orden inverso.

Este proceso de conversión analógico digital o PCM se realiza mediante tres pasos:

- 1.- Muestreo (sampling)
- 2.- Cuantificación (quantization)
- 3.-Codificación (codification)

El proceso de muestreo consiste en tomar valores instantáneos de una señal analógica, a intervalos de tiempo iguales. A los valores instantáneos obtenidos se les llama muestras.

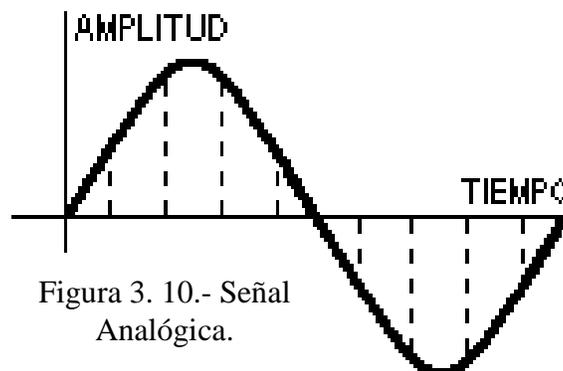


Figura 3. 10.- Señal Analógica.

El muestreo se efectúa siempre a un ritmo uniforme, que viene dado por la frecuencia de muestreo (f_m) o sampling rate.

La condición que debe cumplir el sampling rate viene dada por el teorema del muestreo el cual dice que "Si una señal contiene únicamente frecuencias inferiores a f , queda completamente determinada por muestras tomadas a una velocidad igual o superior a $2f$."

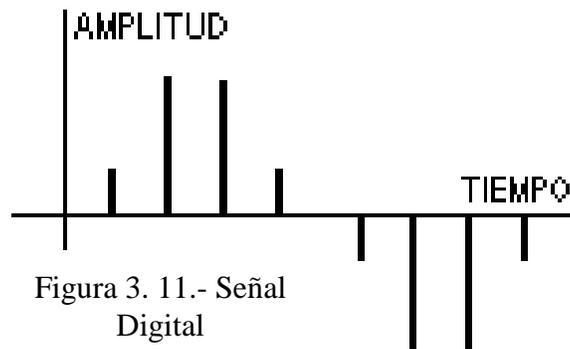


Figura 3. 11.- Señal Digital

De acuerdo con el teorema del muestreo, las señales telefónicas de frecuencia vocal (que ocupan la Banda de 300 a - 3.400 Hz), se han de muestrear a una frecuencia igual o superior a 6.800 Hz (2×3.400).

En la práctica, sin embargo, se suele tomar una frecuencia de muestreo de $f_m = 8.000$ Hz. Es decir, se toman 8.000 muestras por segundo que corresponden a una separación entre muestras de:

$$T=1/8000= 0.000125 \text{ seg.} = 125 \mu\text{s}$$

Por lo tanto, dos muestras consecutivas de una misma señal están separadas $125 \mu\text{s}$ que es el periodo de muestreo.

La cuantificación es el proceso mediante el cual se asignan valores discretos, a las amplitudes de las muestras obtenidas en el proceso de muestreo. Existen varias formas de cuantificar.

Cuantificación uniforme.- Hay que utilizar un número finito de valores discretos para representar en forma aproximada la amplitud de las muestras. Para ello, toda la gama

de amplitudes que pueden tomar las muestras se divide en intervalos iguales y a todas las muestras cuya amplitud cae dentro de un intervalo, se les da el mismo valor.

El proceso de cuantificación introduce necesariamente un error, ya que se sustituye la amplitud real de la muestra, por un valor aproximado. A este error se le llama error de cuantificación.

El error de cuantificación se podría reducir aumentando el número de intervalos de cuantificación, pero existen limitaciones de tipo práctico que obligan a que el número de intervalos no sobrepase un determinado valor.

Una cuantificación de este tipo, en la que todos los intervalos tienen la misma amplitud, se llama cuantificación uniforme.

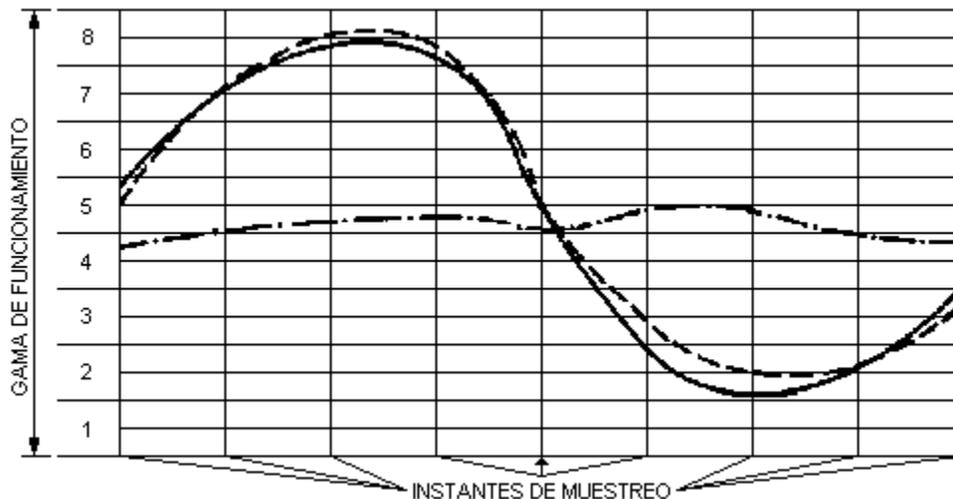


Figura 3. 12.- Cuantificación Uniforme.

Cuantificación no uniforme.- En una cuantificación uniforme la distorsión es la misma cualquiera que sea la amplitud de la muestra. Por lo tanto cuanto menor es la amplitud de la señal de entrada mayor es la influencia del error. La situación se hace ya inadmisibles para señales cuya amplitud analógica está cerca de la de un intervalo de cuantificación.

Para solucionar este problema existen dos soluciones:

- Aumentar los intervalos de cuantificación - si hay más intervalos habrá menos errores pero necesitaremos más números binarios para cuantificar una muestra y por tanto acabaremos necesitando más ancho de banda para transmitirla.
- Mediante una cuantificación no uniforme, en la cual se toma un número determinado de intervalos y se distribuyen de forma no uniforme aproximándolos en los niveles bajos de señal, y separándolos en los niveles altos. De esta forma, para las señales débiles es como si se utilizase un número muy elevado de niveles de cuantificación, con lo que se produce una disminución de la distorsión. Sin embargo para las señales fuertes se tendrá una situación menos favorable que la correspondiente a una cuantificación uniforme, pero todavía suficientemente buena.

Cuantificación diferencial.- En las señales de frecuencia vocal, predominan generalmente las bajas frecuencias, por ello las amplitudes de dos muestras consecutivas difieren generalmente en una cantidad muy pequeña. Aprovechando esta circunstancia, se ha ideado la cuantificación diferencial.

En la cuantificación diferencial, en lugar de tratar cada muestra separadamente, se cuantifica y codifica la diferencia entre una muestra y la que le precede. Como el número

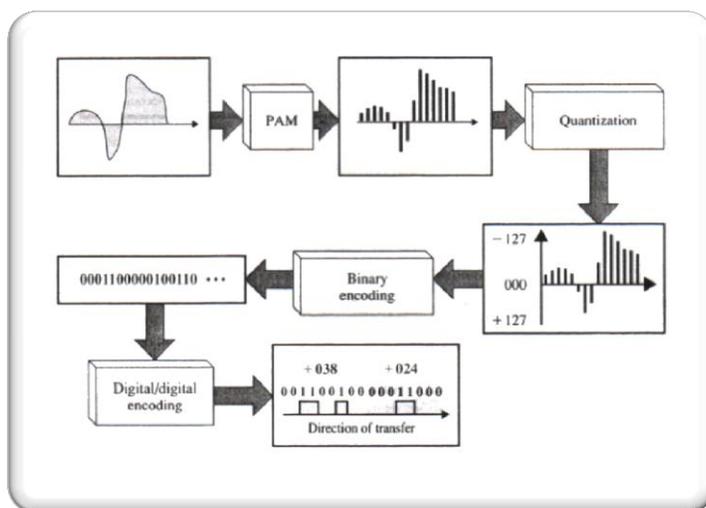


Figura 3. 13.-Transformación de la cuantificación a binario.

de intervalos de cuantificación necesarios para cuantificar la diferencia entre dos muestras consecutivas es lógicamente inferior al necesario para cuantificar una muestra aislada, la cuantificación diferencial permite una reducción sensible de la frecuencia de transmisión en línea, ya que esta es proporcional al número de intervalos de cuantificación.

De lo anterior podemos concluir que la PCM trabaja del siguiente modo:

En la imagen se ve la transformación de una señal analógica a una digital, así como los pasos que se siguen para dicha transformación.

Aquí se muestra la cuantificación para que posteriormente se convierta en señal digital (1 y 0).

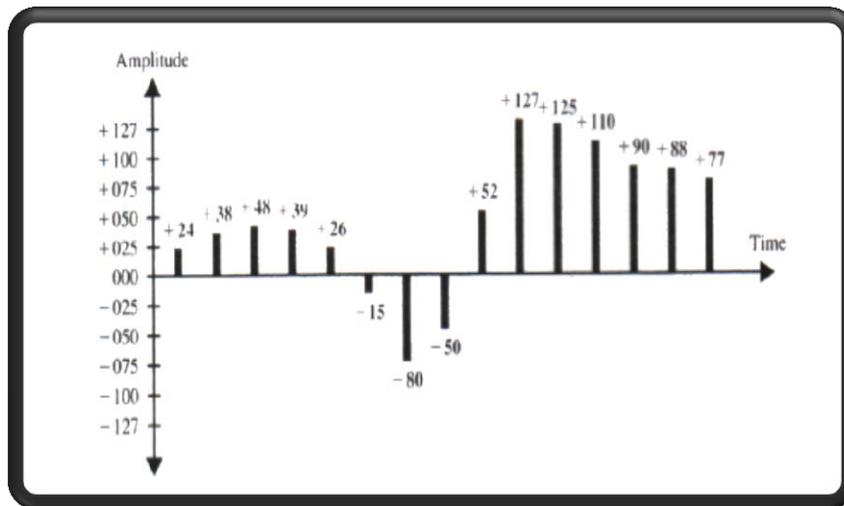


Figura 3. 14.-Cuantificación de la señal analógica.

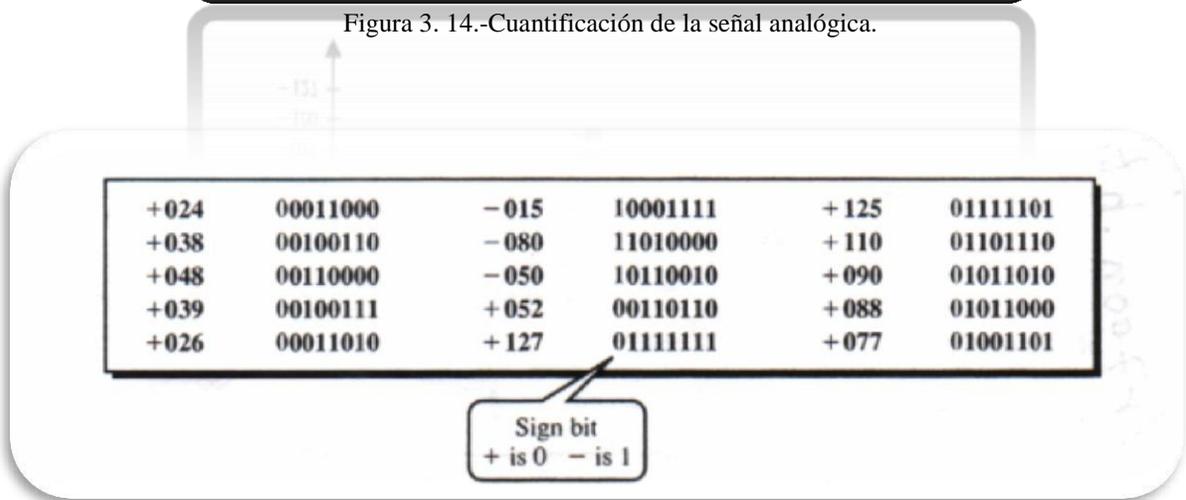


Figura 3. 15.-Transformación de la cuantificación a binario.

A continuación se presentara una tabla comparativa entre los diferentes códec que existen para la conversión de una señal analógica a una digital.

Nombre	Estandarizado	Descripción	Bit Rate (kb/s)	Sampling Rate (kHz)	Frame Size (ms)	Observaciones	MOS (Mean Opinion Score)
G.711	ITU-T	Pulse code modulation (PCM)	64	8	Muestreada	Tiene dos versiones u-law (US, Japan) y a-law (Europa) para muestrear la señal	4.1
G.721	ITU-T	Adaptive differential pulse code modulation (ADPCM)	32	8	Muestreada	Obsoleta. Se ha transformado en la G.726.	
G.722	ITU-T	7 kHz audio-coding within 64 kbit/s	64	16	Muestreada	Divide los 16 KHz en dos bandas cada una usando ADPCM	
G.722.1	ITU-T	Codificación a 24 y 32 kbit/s para sistemas sin manos con baja pérdida de paquetes	24/32	16	20		
G.723	ITU-T	Extensión de la norma G.721 a 24 y 40 kbit/s para aplicaciones en circuitos digitales.	24/40	8	Muestreada	Obsoleta por G.726. Es totalmente diferente de G.723.1.	
G.723.1	ITU-T	Dual rate speech coder for multimedia communications transmitting at 5.3 and 6.3 kbit/s	5.6/6.3	8	30	Parte de H.324 video conferencing. Codifica la señal usando linear predictive analysis-by-synthesis coding. Para el codificador de high rate utiliza Multipulse Maximum Likelihood Quantization (MP-MLQ) y para el de low-rate usa Algebraic-Code-Excited Linear-Prediction (ACELP).	3.8-3.9
G.726	ITU-T	40, 32, 24, 16 kbit/s adaptive differential pulse code modulation (ADPCM)	16/24/32/40	8	Muestreada	ADPCM; reemplaza a G.721 y G.723.	3.85
G.727	ITU-T	5-, 4-, 3- and 2-bit/sample embedded adaptive differential pulse code modulation (ADPCM)	var.		Muestreada	ADPCM. Relacionada con G.726.	
G.728	ITU-T	Coding of speech at 16 kbit/s using low-delay code excited linear prediction	16	8	2.5	CELP.	3.61
G.729	ITU-T	Coding of speech at 8 kbit/s using conjugate-structure algebraic-code-excited linear-prediction (CS-ACELP)	8	8	10	Bajo retardo (15 ms)	3.92
GSM 06.10	ETSI	RegularPulse Excitation LongTerm Predictor (RPE-LTP)	13	8	22.5	Usado por la tecnología celular GSM	
LPC10	Gobierno de USA	Linear-predictive codec	2.4	8	22.5	10 coeficientes.La voz suena un poco "robotica"	
Speex			8, 16, 32	2.15-24.6 (NB) 4-44.2 (WB)	30 (NB) 34 (WB)		
iLBC			8	13.3	30		
DoD CELP	American Department of Defense (DoD) Gobierno de USA		4.8		30		
EVRC	3GPP2	Enhanced Variable Rate CODEC	9,6/4.8 /1.2	8	20	Se usa en redes CDMA	
DVI	Interactive Multimedia Association (IMA)	DVI4 uses an adaptive delta pulse code modulation (ADPCM)	32	Variable	Muestreada		
L16		Uncompressed audio data samples	128	Variable	Muestreada		

Tabla 3. 1.-Características de codec's para la conversión analógico-digital.

Bit Rate.- Indica la cantidad de información que se manda por segundo.

El Sampling Rate indica la frecuencia de muestreo de la señal vocal. (Cada cuanto se toma una muestra de la señal analógica)

El Frame size indica cada cuantos milisegundos se envía un paquete con la información sonora.

El MOS indica la calidad general del códec (valor de 1 a 5)¹⁶

¹⁶ Codec's, <http://www.voipforo.com/codec/codecs.php>. [Consulta: Febrero 2010].

3.1.2.3.-Estándares y Protocolos de señalización IP.

Los protocolos son un conjunto de reglas usadas por computadoras para comunicarse unas con otras a través de una red. Un protocolo es una convención o estándar que controla o permite la conexión, comunicación, y transferencia de datos entre dos puntos finales. En su forma más simple, un protocolo puede ser definido como las reglas que dominan la sintaxis, semántica y sincronización de la comunicación. Los protocolos pueden ser implementados por hardware, software, o una combinación de ambos. A su más bajo nivel, un protocolo define el comportamiento de una conexión de hardware.¹⁷

El objetivo de VoIP es dividir en paquetes los flujos de audio para transportarlos sobre redes basadas en IP. Los protocolos de las redes IP originalmente no fueron diseñados para el fluido en tiempo real de audio o cualquier otro tipo de medio de comunicación. Por tal motivo se crean los protocolos o estándares de comunicación para VoIP, cuyo mecanismo de conexión abarca una serie de transacciones de señalización entre terminales que cargan dos flujos de audio para cada dirección de la conversación.

Uno de los objetivos de cualquier protocolo es de informar al terminal llamado que se desea establecer una comunicación de voz, luego el terminal llamado responderá de alguna forma, aceptando o rechazando dicha comunicación. A este tipo de intercambio de información se la suele llamar señalización de llamada (call signalling).

Hablando de códec existe una gran variedad y hoy en día los más utilizados son G.729, G.711 y GSM. Además en la mayoría de los casos la voz se transporta sobre segmentos UDP, lo que hace necesario la negociación de los puertos UDP donde el receptor espera recibir el audio. Debido a esto, es necesario intercambiar mensajes donde se negocien estas cuestiones y otras más específicas de cada protocolo. Para el intercambio de este tipo de información se definen los protocolos de control de señalización de llamada (Call control signaling) una vez establecida la comunicación, se debe enviar el audio codificado en paquetes IP. Las redes IP suelen tener variaciones de retardo altos respecto a las redes de telefonía tradicionales ya que no fueron diseñadas para el transporte de voz. Y además, por ser una red de datagramas, los paquetes de voz podrían llegar desordenados. Debido a estas características de la red IP, se necesita empaquetar la información de voz

¹⁷ Protocolos, http://es.wikipedia.org/wiki/Protocolo_%28inform%C3%A1tica%29, [Consulta : Febrero 2010]

sobre algún protocolo que minimice o controle estos efectos o defectos. A estos protocolos se los denomina protocolos de transporte de “media” (media transport protocols). Estos protocolos están asociados con los protocolos de control de transporte de “media” (media transport control protocols) cuya función es la de enviar entre los terminales que intervienen en la comunicación, estadísticas sobre jitter (variación en el retardo), paquetes enviados, paquetes recibidos, paquetes perdidos, etc.

Las principales entidades que definen las normas o protocolos de VoIP que hoy se utilizan son la IETF (The Internet Engineering Task Force) por medio de las RFC y la ITU-T (*Unión de telecomunicación internacional*).

3.1.2.4.-Alternativas.

Antes de hablar a ciencia cierta de los protocolos tendré que definir un Protocolo de Tiempo Real llamado *RTP* el cual proporciona los servicios de entrega end-to-end para los datos de características en tiempo real, tales como audio y vídeo. Los servicios incluyen la identificación del tipo de la carga útil, enumeración de la secuencia y monitorización de la entrega. *RTP* está definido sobre UDP así que es sin conexión con entrega con el mejor esfuerzo. Aunque *RTP* es sin conexión, tiene un sistema de secuenciación que permite la detección de paquetes perdidos. Este protocolo no es una alternativa más bien es una característica que nuestras alternativas mencionadas posteriormente tienen consigo, ya que funcionan conjuntamente con los protocolos siguientes.

3.1.2.4.1.-H-323

El estándar H.323 proporciona la base para la transmisión de voz, datos y vídeo sobre redes no orientadas a conexión y que no ofrecen un grado de calidad del servicio, como son las basadas en IP, incluida Internet, de manera tal que las aplicaciones y productos puedan interoperar, permitiendo la comunicación entre los usuarios sin necesidad de que éstos se preocupen por la compatibilidad de sus sistemas. La LAN sobre la que los terminales H.323 se comunican puede ser un simple segmento o un anillo, o múltiples segmentos (es el caso de Internet) con una topología compleja, lo que puede resultar en un grado variable de rendimiento.

Fue creado y establecido por la UIT (Unión Internacional de Telecomunicaciones) en el año de 1996, que fija los estándares para la comunicación de voz y vídeo sobre redes de área local. Para la conferencia de datos se apoya en la norma T.120, con lo que en conjunto soporta las aplicaciones multimedia. Los terminales y equipos conforme a H.323 pueden tratar voz en tiempo real, datos y vídeo.

El estándar contempla el control de la llamada, gestión de la información y ancho de banda para una comunicación punto a punto y multipunto, dentro de la LAN, así como define interfaces entre la LAN y otras redes externas, como puede ser la RDSI.

En este protocolo se establecen los estándares para la compresión y descompresión de audio y vídeo, asegurando que los equipos de distintos fabricantes se entiendan. Así, los usuarios no se tienen que preocupar de cómo el equipo receptor actúe, siempre y cuando cumpla este estándar. La gestión del ancho de banda disponible para evitar que la LAN se sature con la comunicación de audio y vídeo, por ejemplo, limitando el número de conexiones simultáneas, también está contemplada en el estándar.

El protocolo H.323 hace uso de los procedimientos de señalización de los canales lógicos contenidos en la norma H.245, en los que el contenido de cada uno de los canales se define cuando se abre. Estos procedimientos se proporcionan para fijar las prestaciones del emisor y receptor, el establecimiento de la llamada, intercambio de información, terminación de la llamada y como se codifica y decodifica.

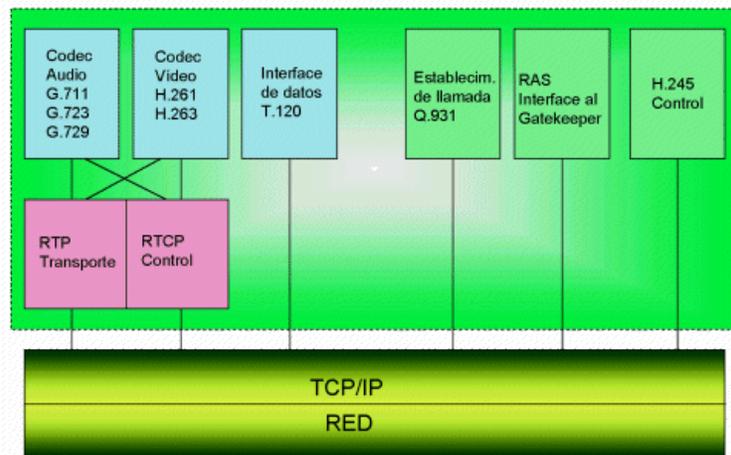


Figura 3. 16.-Arquitectura H323.

3.1.2.4.2.-MGCP

MGCP (Media Gateway Control Protocol). Este protocolo interno fue desarrollado principalmente para hacer frente a las exigencias de la compañía basada en redes de telefonía IP. MGCP es un protocolo complementario de ambos H.323 y SIP, que fue diseñado como un protocolo interno entre el Media Gateway Controller y el Media Gateway.

MGCP está compuesto por una llamada de agente, un MG (Media Gateway), que realiza la conversión de las señales de los medios de comunicación entre circuitos y paquetes, y una SG (puerta de enlace de señalización) cuando se conecta a la PSTN (Red Telefónica Pública Conmutada). MGCP es ampliamente utilizado entre los elementos de una puerta de enlace multimedia descompuesto. El portal tiene una llamada de agente, que está compuesto de control de la llamada "inteligencia" y una puerta de entrada cuenta con los medios de comunicación las funciones de los medios de comunicación, por ejemplo la conversión de voz TDM a voz sobre IP.

En el protocolo MGCP las conexiones pueden ser punto a punto o multipunto. Conexión punto a punto puede ser una conexión entre dos puntos finales para la transmisión de datos entre estos extremos. Una vez que se configura la conexión entre dos extremos, la transferencia de datos tiene lugar entre los extremos. En una conexión multipunto, la conexión se establece entre un extremo y un período de sesiones multipunto. En una conexión multipunto, las conexiones se pueden crear más de portador de diversos tipos de redes.

MGCP en la arquitectura, todos y cada uno cuenta con un comando ID de transacción, se recibe un reconocimiento y una respuesta. Esto puede ser mejor entendido como la arquitectura de suscripción, como la convocatoria de agente informa a los medios de comunicación y la señalización de puertas de entrada como puertas de entrada a lo que los acontecimientos que se ocupa de los acontecimientos y lo que deja desatendida.

3.1.2.4.3.-MEGACO

Megaco (control de gateway de medios) es un protocolo del control y de señalización para las conexiones de VOIP. Megaco fue desarrollado en común por el IETF y la ITU. H.248 es el nombre de ITU para Megaco, este fue desarrollado apartir de MGCP y por lo tanto es muy similar. Los protocolos como Megaco y MGCP se utilizan cuando la función del Media Gateway es separada del Media Gateway Controller/Softwitch. Megaco es utilizado por el MGC (Controlador de la pasarela de medios) para iniciación y control de media streams (voz) a través del MG (Pasarela de medios).

Una de las características fundamentales de este modelo, es que los Media Gateway son capaces (en teoría) de mantener comunicaciones tanto con el H.323 como con el protocolo SIP, algo fundamental para la óptima implantación del sistema VoIP.

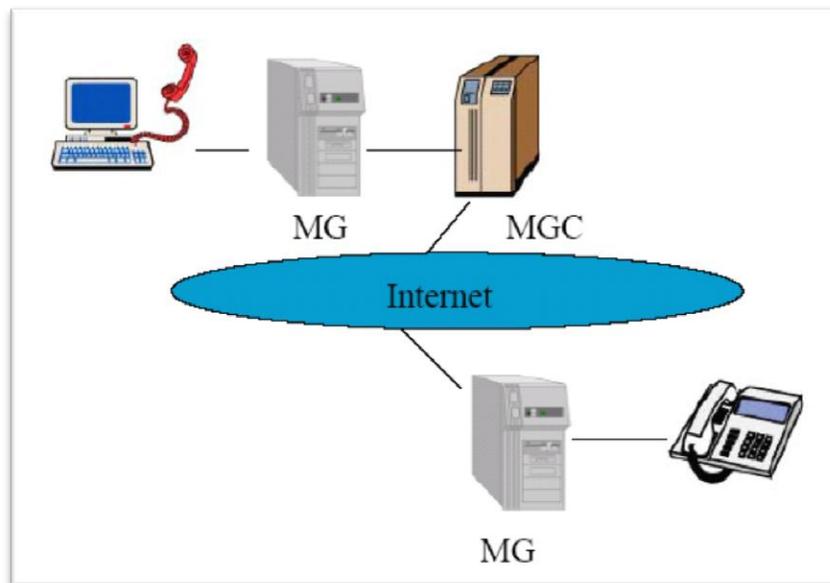


Figura 3. 17.-Esquema del protocolo MEGACO.

Recordemos que la comunicación, puede realizarse bien sea a través de un ordenador con un altavoz y un micrófono, o bien a través de un teléfono analógico. Si un usuario desea realizar una llamada, mediante este sistema, los pasos que se realizan son los siguientes:

- 1) El usuario descuelga el teléfono y marca el número de teléfono del destinatario. Esta llamada, le llega al Media Gateway.
- 2) El Media Gateway, notifica al Media Gateway Controller de que una llamada está en camino.
- 3) El Media Gateway Controller busca en su base de datos, el número de teléfono del destinatario para saber su IP y su número de puerto. Entonces, busca el Media Gateway del destinatario, y le envía un mensaje para indicarle que le está llegando una llamada.
- 4) El Media Gateway del destinatario, abre una RTP (Protocolo en tiempo real) cuando el usuario descuelga.

3.1.2.4.4.-SIP

El protocolo de iniciación de sesión (SIP) es un protocolo de señalización para crear, modificar, y terminar sesiones con unos o más participantes. Estas sesiones incluyen llamadas telefónicas por Internet, distribución de datos multimedia, y conferencias multimedia.

El SIP hace uso de elementos llamados servidores Proxy para ayudar a encaminar peticiones a la localización actual del usuario, a autenticar y a autorizar a usuarios para los servicios, implementar políticas de encaminamiento, y proporcionar servicios a los usuarios. El SIP también proporciona una función de registro que permite que los usuarios indiquen sus localizaciones actuales para ser usadas por los servidores Proxy. SIP funciona por encima de varios protocolos del transporte como RTP/RTCP.

SIP tiene mucha similitud con HTTP, el protocolo de Web, o SMTP. Los mensajes consisten de encabezados y un cuerpo de mensaje como HTTP. Los cuerpos de mensaje de SIP para las llamadas telefónicas se definen en SDP (protocolo de descripción de la sesión). El SIP ofrece todas las potencialidades y las características comunes de la telefonía de Internet como:

- llamada o transferencia de medios
- conferencia de llamada
- llamada en espera,etc.

3.1.2.4.5.-IAX2

Este protocolo ha sido desarrollado para solucionar problemas de NAT (Network Address Translation) y mejorar el trunking entre sistemas basados en este protocolo. En la comunicaciones basadas en IAX, el Asterisk puede operar de dos formas diferentes: Servidor: Como Servidor, Asterisk admite registros de clientes IAX, pudiendo ser estos clientes Hardware, Software u otros Asterisk.

Cliente: Como Cliente, Asterisk puede registrarse en otros Asterisk o en operadores IP que utilicen este protocolo.

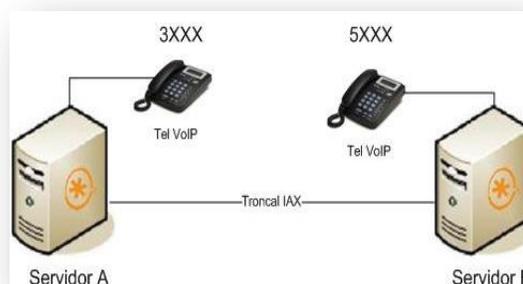


Figura 3. 18.-Esquema de conexión IAX2.

Este protocolo, que fue diseñado por Mark A. Spencer, Brian Capouch, Ed Guy y Frank Miller, permite conectar a través de paquetes UDP dos sistemas o terminales hacia un PBX (central telefónica) con Asterisk. Lo relevante es que su publicación como RFC (Request For Comments) permitirá una adopción del mismo a niveles como SIP o H323 en aplicaciones de voz sobre IP.

IAX2 utiliza un único puerto UDP, generalmente el 4569, para comunicaciones entre puntos finales (terminales VoIP) para señalización y datos. El tráfico de voz es transmitido in-band, lo que hace a IAX2 un protocolo casi transparente a los cortafuegos y realmente eficaz para trabajar dentro de redes internas. En esto se diferencia de SIP, que utiliza una cadena RTP out-of-band para entregar la información.

El diseño de IAX se basó en muchos estándares de transmisión de datos, incluidos SIP (el cual es el más común actualmente), MGCP y Real-time Transport Protocol.

El principal objetivo de IAX ha sido minimizar el ancho de banda utilizado en la transmisión de voz y vídeo a través de la red IP, con particular atención al control y a las llamadas de voz y aparte brindar un soporte nativo para ser transparente a NAT. La estructura básica de IAX se fundamenta en la multiplicación de la señalización y del flujo de datos sobre un simple puerto.

3.1.3.-Ventajas de protocolos.

El siguiente cuadro trata de realizar una comparación entre las características más importantes de los protocolos para VoIP antes descritos:

	Tecnología	Disponibilidad	Seguridad	NAT	Total
SIP	2	2	2	1	7
IAX	2	3	1	3	9
H.323	3	1	2	1	7
MGCP	2	1	¿?	¿?	3
MEGACO	3	1	¿?	¿?	4

Tabla 3. 2.-Comparación de ventajas.

Con relación al ancho de banda IAX utiliza un menor ancho de banda que SIP ya que los mensajes son codificados de forma binaria mientras que en SIP son mensajes de texto. Asimismo, IAX intenta reducir al máximo la información de las cabeceras de los mensajes reduciendo también el ancho de banda.

Otra cuestión que se podría tomar como ventaja de IAX2 es que la señalización y los datos viajan conjuntamente con lo cual se evitan los problemas de NAT.

En cuanto a SIP al ser un protocolo estandarizado por la IETF hace bastante tiempo y que es ampliamente implementado por todos los fabricantes de equipos y software tiene una gran ventaja en contra de los demás protocolos. IAX al estar siendo estandarizado y no tener una madures como SIP, no se encuentran muchos dispositivos existentes en el mercado.

IAX utiliza un solo puerto para mandar la información de señalización y los datos de todas sus llamadas. SIP utiliza un puerto para señalización y 2 puertos RTP por cada conexión de audio (como mínimo 3 puertos). Por ejemplo para 100 llamadas simultáneas

con SIP se usarían 200 puertos (RTP) más el puerto 5060 de señalización. IAX utilizaría sólo un puerto para todo.

3.1.4.- Protocolo elegido para la implementación en Luz y fuerza y ¿Porque?

SIP o Protocolo de Iniciación de Sesión es un estándar genérico que permite compatibilidad con otros productos basados en SIP ofreciendo flexibilidad y opciones para crear soluciones que de abasto para sus propias necesidades.

Además de que todos los fabricantes de hardware tienen soluciones económicas para poner en marcha una PBX con Asterisk. Y además es un estándar sólido.

En luz y Fuerza lo que se debe de tomar en cuenta para elegir un protocolo es el soporte y al llevar más tiempo y ser muy estable podremos tomar la decisión de elegir SIP como el protocolo que utilizaremos para este proyecto.

3.1.5.- Protocolo SIP

El 22 de febrero de 1996 Mark Handley y Eve Schooler presentaron al IETF un borrador del Session Invitation Protocol conocido ahora como SIPv1. El mismo estaba basado en trabajos anteriores de Thierry Turletti (INRIA Videoconferencing System o IVS) y de Eve Schooler (Multimedia Conference Control o MMCC).

Ese mismo día el Dr. Henning Schulzrinne presentó un borrador del Simple Conference Invitation Protocol (SCIP), que estaba basado en el HTTP (Hypertext Transport Protocol). Usaba TCP (Transmission Control Protocol) como protocolo de transporte. Como identificadores de los usuarios utilizaba direcciones de correo electrónico para permitir el uso de una misma dirección para recibir correos electrónicos e invitaciones a conferencias multimedia.

El IETF decidió combinar ambos en un único protocolo denominado Session Initiation Protocol, (es decir cambiando el significado de la inicial I en el acrónimo "SIP") y su número de versión fue el dos, dando origen al SIPv2. En diciembre de 1996 los tres autores (Schulzrinne, Handley y Schooler), presentaron el borrador del SIPv2. El mismo luego de ser discutido en el grupo de trabajo MMUSIC (Multiparty Multimedia Session Control) del IETF alcanzó el grado de "proposed standard" en la RFC 2543 publicada en

febrero de 1999. En septiembre de 1999 se creó el grupo de trabajo SIP en el IETF que continuó con el desarrollo del protocolo y en junio de 2002 se publicó la RFC 3261 que reemplazó a la anterior introduciendo modificaciones propuestas durante el trabajo del grupo SIP.

El protocolo de inicio de sesiones (SIP, SessionInitiationProtocol) es un protocolo de señalización de capa de aplicación que define la iniciación, modificación y la terminación de sesiones interactivas de comunicación multimedia entre usuarios.

Este protocolo utiliza aparte otro protocolo, el **RTP** para la transmisión en tiempo real de video y audio.

Este protocolo es también conocido como un protocolo de requerimiento-respuesta. La sintaxis de sus operaciones como ya se dijo se asemeja a las de HTTP, los protocolos utilizados en los servicios de páginas Web. Esta similitud es natural ya que SIP fue diseñado para que la telefonía se vuelva un servicio más en la Internet.

Destacamos que los mensajes SIP constan de texto legible, permitiendo que los humanos puedan analizarlos fácilmente.

“El uso de SIP es cada vez más predominante en el mundo de los protocolos: una clara señal de su buen diseño.”¹⁸

SIP fue creado únicamente para cumplir con la función de establecer una sesión, sin importar sus características internas. Sólo se encarga de comenzar, terminar, y modificar sesiones.

SIP fue creado únicamente para cumplir con la función de establecer una sesión, sin importar sus características internas. Sólo se encarga de comenzar, terminar, y modificar sesiones.

SIP se ayuda de otros protocolos para lograr una llamada telefónica, una sesión de video conferencia o de Mensajería Instantánea, etc. Los protocolos que apoyan comúnmente a SIP son: SDP y RTP (RTCP). RTP es usado para transportar los datos multimedia en tiempo real mientras que SDP se emplea para describir y codificar las características y capacidades de los participantes en la sesión.

¹⁸ SIP, <http://objetoide.blogspot.com/2008/06/protocolo-sip-qu-es-y-para-qu-sirve.html>, [Consulta : Febrero 2010].

Sus principales características y funciones del protocolo son:

- Establecer, modificar y finalizar sesiones entre dos o más terminales.
 - Sintaxis similar a HTTP o SMTP.
 - Uso de URIs (con esquemas sip, sips y tel).
 - Métodos básicos: INVITE, ACK, BYE, CANCEL, REGISTER, OPTIONS.
 - Los mensajes se agrupan en transacciones y llamadas.
 - Generalmente, el cuerpo de los mensajes contiene descripciones de sesiones multimedia (SDP).
 - Códigos de respuesta similares a los de HTTP. (Ejemplo: 200 – OK)
 - Localización basada en DNS.
 - Registro y localización de participantes. Movilidad.
 - Gestión del conjunto de participantes y de los componentes del sistema.
 - Descripción de características de las sesiones y negociación de capacidades de los participantes.
- Algunas de sus características son:
- Basado en Texto
 - Cabeceras como método de ampliación.

Tabla 3. 3.-Características y funciones del protocolo SIP.

3.1.5.1.-Diseño del protocolo

El protocolo SIP fue diseñado con el concepto de "caja de herramientas", es decir, el protocolo SIP se vale de las funciones aportadas por otros protocolos, las que da por hechas y no vuelve a desarrollarlas. Debido a este concepto SIP funciona en colaboración con otros muchos protocolos. El protocolo SIP se concentra en el establecimiento, modificación y terminación de las sesiones, se complementa, entre otros, con el SDP, que describe el contenido multimedia de la sesión, por ejemplo qué direcciones IP, puertos y códecs se usarán durante la comunicación. También se complementa con el RTP (Real-time Transport Protocol). RTP es el verdadero portador para el contenido de voz y video que intercambian los participantes en una sesión establecida por SIP.

El protocolo SIP adopta el modelo cliente-servidor. El cliente realiza peticiones (requests) que el servidor atiende y genera una o más respuestas (dependiendo de la naturaleza, método o petición).

Otro concepto importante en el diseño de este protocolo es el de la extensibilidad. Esto significa que las funciones básicas del protocolo, definidas en la RFC 3261, pueden ser extendidas mediante otras RFC (Requests for Comments) dotando al protocolo de funciones más potentes. En otras palabras el protocolo SIP puede ser dotado de más funcionalidades trabajando junto con otros protocolos para realizar tareas más complejas.

3.1.5.2.-Funcionamiento del protocolo SIP

SIP fue desarrollado por la IETF (RFC3261), por lo tanto su desarrollo está orientado a la integración con aplicaciones y servicios de Internet. Tiene mayor flexibilidad para incorporar nuevas funciones y su implementación es más simple.

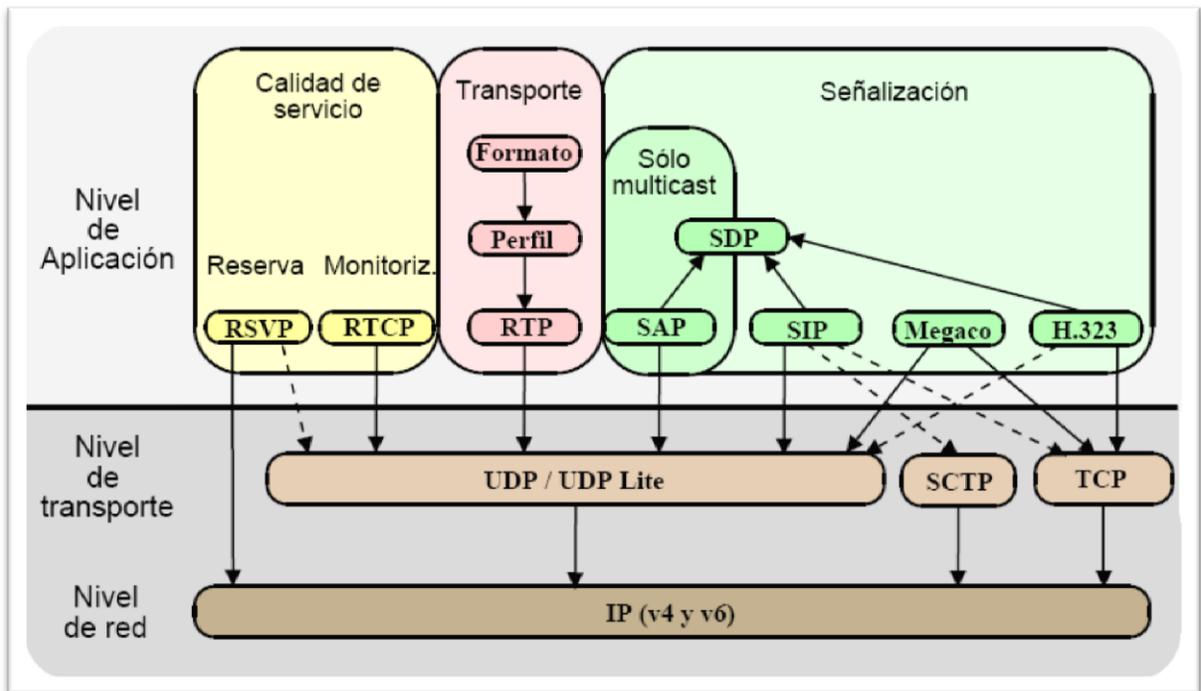


Figura 3. 19.-Esquema de funcionamiento del protocolo SIP.

SIP es un protocolo de la capa de Aplicación. Como se puede observar, está relacionado estrechamente con el Protocolo SDP y coexiste junto con otros protocolos del mismo nivel y funciones, como los mencionados con anterioridad y que por sus características no se eligieron para este proyecto.

3.1.5.2.1.-Agentes de usuario

Los usuarios, que pueden ser seres humanos o aplicaciones de software, utilizan para establecer sesiones lo que el protocolo SIP denomina "Agentes de usuario". Estos no son más que los puntos extremos del protocolo, es decir son los que emiten y consumen los mensajes del protocolo SIP. Un softphone, un teléfono IP o cualquier otro dispositivo similar es para este protocolo un agente de usuario. El protocolo SIP no se ocupa de la interfaz de estos dispositivos con el usuario final, sólo se interesa en los mensajes que estos generan y cómo se comportan al recibir determinados mensajes.

Todos los terminales de Telefonía IP tienen cierta inteligencia que está proporcionada por un agentes de usuario (UA), se comportan como clientes (UAC: User Agent Clients) y como servidores (UAS: User Agent Servers). Son UAC cuando realizan una petición y son UAS cuando la reciben. Estos agentes se consisten en un programa informático (siempre activo, del tipo demonio).

Cuando un UAC emite una solicitud, esta pasa por algunos proxys y termina en un UAS del corresponsal. Cuando dicho UAS responde, la respuesta se hace llegar al UAC del otro extremo de la conversación. Los procedimientos UAC y UAS dependen de dos factores. Si la solicitud o respuesta forma parte o no de un diálogo, y cuál es el método invocado por la solicitud.

Algunos de los dispositivos que pueden tener una función de UA son los Teléfonos IP, ordenadores con Softphone, pasarelas telefónicas y sistemas de mensajería automática.

Un Agente de Usuario implica que el aparato donde esté instalado tiene cierta capacidad de proceso, para poder actuar con UAC y UAS e interactuar con el usuario.

3.1.5.2.2.-Servidores de Registro

El protocolo SIP permite establecer la ubicación física de un usuario determinado, esto significa que se sabe en qué punto de la red está conectado cierto usuario terminal ip. Para ello se vale del mecanismo de registro, el cual sigue los siguientes pasos:

Cada usuario tiene una dirección lógica que es invariable respecto de la ubicación física del usuario. La forma de identificar a una entidad SIP es similar a la empleada para definir una cuenta de correo electrónico. A esta forma se le denomina URI (Uniform Resource Identifier). El URI de SIP es de la forma sip:usuario@dominio, por ejemplo: sip:usuario@lyf.gob.mx. La dirección física (denominada "dirección de contacto") es dependiente del lugar en donde el usuario está conectado (de su dirección IP).

Cuando un usuario inicializa su terminal (por ejemplo conectando su teléfono o abriendo su software de telefonía SIP) el agente de usuario SIP que reside en dicho terminal envía una petición con el método REGISTER a un Servidor de Registro, informando a qué dirección física debe asociarse la dirección lógica del usuario. El servidor de registro realiza entonces dicha asociación (denominada binding). Esta asociación tiene un período de vigencia y si no es renovada, caduca. También puede terminarse mediante un desregistro.

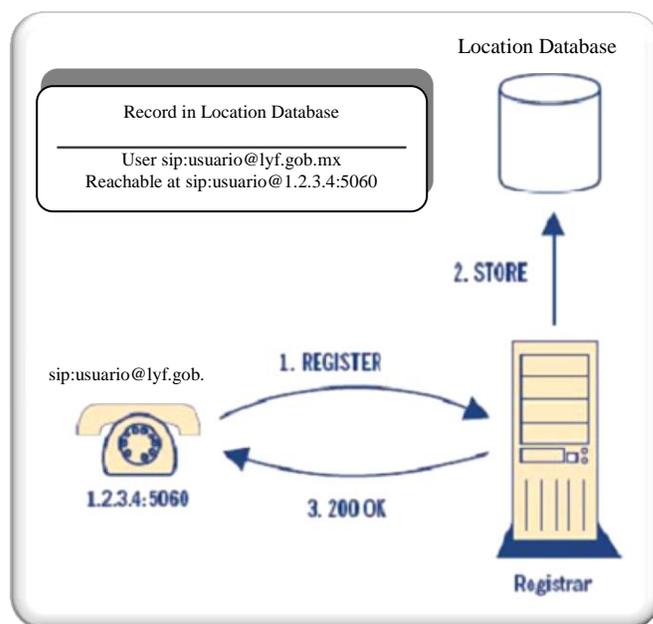


Figura 3. 20.-Funcionamiento de los servidores de registro.

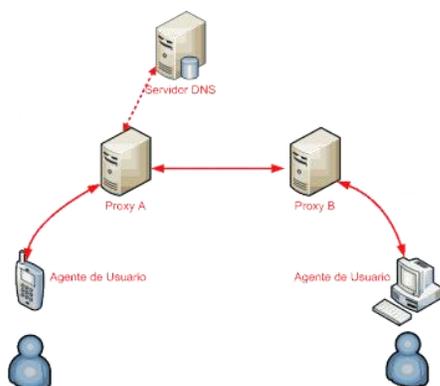
3.1.5.2.3.-Servidores Proxy

El término Proxy significa "Poder de representación". Frente a una solicitud actúa como si fuese un servidor devolviéndonos una respuesta, pero realmente en muchas ocasiones tendrá que consultar a otros proxys, ante quienes será un cliente que solicita datos. Una vez que obtenga lo solicitado, nos lo presentará. Por tanto, en ese momento ha terminado de hacer el recado. Durante el proceso ha actuado tanto como cliente (frente a otros proxys) como servidor (frente a nosotros).

Los servidores proxy se suelen utilizar en funciones de enrutado, por tanto su función principal es conseguir que la solicitud del cliente se remita a la entidad más cercana al usuario de destino. Pero los Proxys también se emplean para verificar las políticas (esto es, comprobar si el usuario está autorizado a efectuar una llamada). El proxy interpreta, y si fuese preciso, reescribe partes del mensaje de solicitud antes de reenviarlo.

Por tanto, de manera sencilla se puede decir que un proxy es un programa que actúa como servidor o accesa a otros servidores. Es un punto intermedio que sirve para aumentar la seguridad o la velocidad de acceso (o el anonimato). Suele tener lo que se conoce como cache, que es una copia temporal de los datos que ha obtenido recientemente.

En la señalización SIP los agentes de usuario (UA), y los elementos intermedios como "Servidores Proxy".



En el ejemplo: un llamante en el dominio A (UA1) quiere llamar a un usuario en el dominio B (usuario@B). Para hacerlo, se comunica con el proxy A de su dominio. El Proxy A remite la solicitud al proxy del dominio de la persona llamada (dominio B), que es el proxy B. Y el Proxy B remite la llamada al Agente de Usuario de la persona llamada, UA 2.

Figura 3. 21.-Ejemplo de servidores proxy.

3.1.5.2.4.-Servidores de Redirección

En algunas arquitecturas es conveniente reducir la carga del Servidor Proxy y mejorar la robustez de la señalización mediante un servidor de redirección.

Un Servidor de Redirección acepta solicitudes SIP y mapea o convierte la dirección SIP de la persona llamada a cero o más direcciones y las devuelve al cliente que las solicitó. A diferencia de los servidores Proxy, los Servidores de Redirección no pasan la solicitud a otros servidores.

El servidor de Redirección contestará a una consulta del cliente devolviendo la información de enrutado sin atender más mensajes de la transacción. Cuando quien haya originado la solicitud reciba la redirección, hará una nueva solicitud basada en la URI que ha recibido.

3.1.5.2.5.-Formato de los mensajes.

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

Otra parte importante son las URI's que son el esquema de direccionamiento SIP para llamar a otra persona vía SIP. En otras palabras, un SIP URI es como un número telefónico SIP de un usuario. El URI es parecido a una dirección de correo electrónico y es escrito en el siguiente formato:

SIP URI = sip:x@y:Puerto

Donde x=Nombre de usuario y y=equipo (dominio o IP).

3.1.5.2.6.- Métodos (Solicitudes).

Solicitudes SIP: estos mensajes son emitidos por el UAC y constan básicamente de tres bloques: Request Line + Cabeceras + Cuerpo. La línea Request Line tiene el siguiente formato:

Método SP **Request-URI** SP **SIP-Versión** CRLF

Donde SP es el carácter espacio, y CRLF es retorno de carro.

Metodo	Descripcion
INVITE	Solicita el inicio de una sesión
ACK	Confirma que se ha iniciado una sesión
BYE	Solicita la terminación de una sesión
OPTIONS	Consulta a un host sobre sus capacidades
CANCEL	Cancela una solicitud pendiente
REGISTER	Informa al servidor de re direccionamiento sobre la ubicación actual del usuario

Tabla 3. 4.-Métodos (Solicitudes).

3.1.5.2.7.-Códigos de Estado (Respuestas).

Respuestas SIP: estos mensajes son emitidos por los UAS's o servidores y se utilizan para responder a un mensaje de solicitud SIP, constan de tres bloques al igual que los mensajes de solicitud: Status Line + Cabeceras + Cuerpo. La línea Status Line tiene el siguiente formato:

SIP-Versión SP **Status-Code** SP **Reason-Phrase** CRLF

Codigo	Significado	Ejemplos
1xx	Información	100=El servidor está de acuerdo en manejar la solicitud del cliente
2xx	Éxito	200=la solicitud es exitosa;204=no hay contenido
3xx	Redirección	301=página movida;304=página no encontrada
4xx	Error del Cliente	403=página prohibida;404=página no encontrada
5xx	Error del Servidor	500=error interno del servidor;503=trata más tarde

Tabla 3. 5.-Códigos de estado (respuestas).

Cabe señalar, como fue mencionado anteriormente, que para ambos tipos de mensajes el cuerpo del mensaje es opcional, pudiendo tenerlo o no.

3.1.5.2.8.-Flujo de establecimiento de una sesión.

El establecimiento de una sesión en SIP es una secuencia de mensajes entre dos elementos de Red. Una sesión corresponde a una petición y todas las respuestas a esa petición. Esto quiere decir que un establecimiento de sesión incluirá cero o más respuestas provisionales y una o más respuestas finales. Las entidades SIP que almacenan el estado de las transacciones son denominadas Stateful. Lo hacen por medio del registro de cada transacción a través de un identificador contenido en el encabezado.

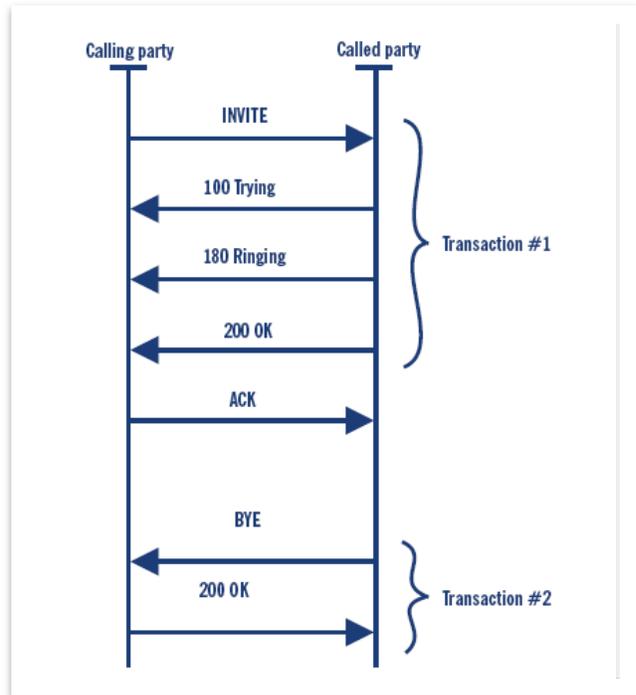


Figura 3. 22.-Flujo de sesión

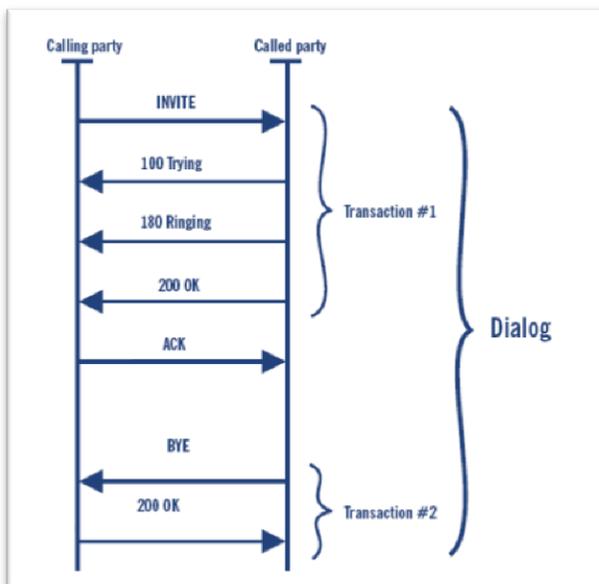


Figura 3. 23.-Diálogo (peer to peer).

El establecimiento de una sesión se da por un diálogo SIP es una conversación peer-to-peer entre dos UA. Los diálogos son identificados usando los campos Call-ID (Id. De llamada), From (de) y To (para).

La conexión se realiza utilizando un acuerdo de 3 vías, de modo que el invocador responde con un mensaje ACK para terminar el protocolo y confirmar la recepción del mensaje del código de respuesta 200.

Cualquiera de las 2 partes puede solicitar la terminación de una sesión enviando un mensaje

que contiene el método BYE. Cuando el otro lado confirma su recepción, se termina la sesión (siempre en 3 vías).

3.2.-Calidad del Servicio (QoS).

Para que las redes aseguren nuestra información se requieren mecanismos que nos aseguren que la voz tendrá prioridad, esto debido a que las conversaciones telefónicas ocurren en tiempo real, de manera que es inaceptable que los paquetes de VoIP lleguen tarde o demasiado tarde o no lleguen nunca.

Para resolver este problema se han creado mecanismos para asegurarnos de que los paquetes de VoIP sean priorizados dentro de una red, lo que se denomina calidad de servicio (QoS).

3.2.1.-Factores que impactan la calidad del servicio.

Los factores que intervienen en la calidad de la voz son los siguientes:

- Códec
- Ancho de Banda
- Latencia (Retardo en la llegada de paquetes)
- Jitter (Variación de la latencia)
- Perdida de Paquetes
- Eco

Es común que la gente piense que con solo aumentar el ancho de banda a la VoIP se solucionarían sus problemas, pero esto no es necesariamente cierto.

Supongamos que en Luz y Fuerza es común utilizar teléfonos IP que integran dos puertos Ethernet, de modo que se comparte un único punto en la red entre su teléfono IP y su computadora. Ahora bien, supongamos además que usted necesita transferir archivos con frecuencia entre su PC y otros equipos de la red, esta transferencia es como una ‘avalancha’ de datos que tiende a consumir todo el ancho de banda disponible en su conexión a la red lo cual afecta gravemente el rendimiento de aplicaciones de audio o de video.

La capacidad para permitir la pérdida de paquetes en la telefonía IP es muy baja, y debe mantenerse por debajo del 2% para que no afecte la calidad de del mensaje de voz. Pero los problemas que en realidad más afectan a la telefonía IP son la Latencia, el Jitter y el ECO.

3.2.1.1.-Latencia.

Se define como el retardo entre el envío de paquetes y su recepción, pero desde un punto de vista del usuario se definirá como el vacío en la conversación producido por retardos acumulados durante la transferencia de paquetes y los retardos de procesamiento. En redes locales donde el ancho de banda no es una limitante no hay problema, pero en conexiones que usan Internet se requiere un compresor para disminuir el ancho de banda utilizado por cada llamada.

3.2.1.2.-Jitter.

El Jitter es la variación en los retardos en la llegada de los paquetes entre su origen y el destino usualmente producida por congestión de tráfico en algún punto de la red o diferencia en el tiempo de transito de paquetes cuando estos viajan por diferentes rutas.

La solución usual es utilizar un Buffer (Jitter Buffer) que almacene los paquetes antes de entregarlos al destino asegurándose así de que lleguen todos en orden al destino, aunque esto introduce un retardo adicional. En aplicaciones de Telefonía IP como Asterisk es posible configurar un jitter buffer para corregir este problema.

3.2.1.3.-Eco.

Cuando tenemos latencia y jitter en una comunicación por VoIP se puede producir el fenómeno del Eco lo que por lo general se resuelve usando ‘canceladores de eco’, ya sean por software o por hardware.

Existen 2 tipos de eco comunes: uno de alto nivel y poco retardo que se produce a nivel local y otro de bajo nivel y mayor retardo que se produce en el extremo remoto. Ambos pueden ser removidos por medio de los canceladores de eco pero es importante comprender que por lo general los canceladores de eco por hardware son más eficientes que aquellos

construidos por software, regla que aplica principalmente a aplicaciones de Telefonía IP por Software como Asterisk,

Por otro lado algunas veces el eco se produce dentro del mismo teléfono IP por un pobre diseño acústico o por influencia directa de los materiales con los que se ha construido, lo que puede afectar el desempeño de teléfonos IP de bajo costo que aunque cuentan con una electrónica de calidad son afectados a nivel acústico al permitir un retorno de la voz entre el auricular y el micrófono. Por lo general este tipo de eco no puede ser resuelto con canceladores de eco externos al teléfono.

3.2.1.4.-Pérdida de Paquetes.

El porcentaje de paquetes que no llegan a su destino mide la pérdida de paquetes de la red. Esta pérdida puede producirse por errores en alguno de los equipos que permiten la conectividad de la red o por sobrepasar la capacidad de algún buffer de algún equipo o aplicación en momentos de congestión. Normalmente en aplicaciones que no funcionan en tiempo real pueden aprovecharse de la retransmisión de los paquetes, pero la telefonía IP funcionan en tiempo real y sus paquetes no pueden ser retransmitidos.

3.2.2.- Conversión análoga digital.

Tenemos que evaluar tecnologías, los servicios de los que disponen éstas tecnologías, en la selección del modelo y la tecnología que mejor se adapte a las necesidades del proyecto en Luz y Fuerza así como también los objetivos definidos para la red.

Los sistemas de voz paquetizados siguen un modelo común. Las redes de transporte de paquetes que pueden estar basadas en IP, Frame Relay o ATM.

Los llamados agentes de voz que son los dispositivos o componentes que forman parte de estas redes y cuya misión principal es la de cambiar la información de la voz desde la forma tradicional de telefonía a una forma más fluida para la transmisión de paquetes, las redes tienden a pasar de lo que es los paquetes de datos a los llamados agentes de voz sirviendo las llamadas.

Hay tres tecnologías para convertir la información en paquetes.

- Voz sobre ATM (VoATM)
- Voz sobre Frame Relay (VoFR)
- Voz sobre IP (VoIP)

3.2.2.1.-Voz sobre Frame Relay

La Voz sobre Frame Relay fue diseñada inicialmente para aplicaciones corporativas en las que las empresas con oficinas remotas internacionales se interconectan mediante líneas Frame Relay infrautilizadas. Estas empresas pueden añadir voz y servicios de fax fácilmente utilizando un gateway VoFR en su ubicación central y en las remotas.

En 1997, el Frame Relay Forum estandarizó por primera vez la voz y el fax sobre redes Frame Relay en su recomendación FRF.11, que está diseñada para comunicaciones punto a punto entre dos gateways VoFR, pero no se enfrenta a los problemas con las redes telefónicas conmutadas públicas. Además, este primer estándar está diseñado sólo para soportar llamadas de voz y fax por Circuitos Virtuales Permanentes. Para soportar voz y fax sobre Circuitos Virtuales Conmutados (SVCs) se ha tenido que recurrir a los esquemas de señalización específicos de cada fabricante, lo que introduce un factor de dependencia de un solo suministrador.

En cuestión de ventajas hay que destacar tiene la capacidad de establecer conexiones virtuales que siguen un camino lógico, mejor que limitarse a seguir una ruta física predefinida a través de la red, una prestación muy indicada para aplicaciones como centros de llamadas distribuidos.

3.2.2.2.-Voz sobre ATM

Voz sobre ATM es la solución más sofisticada para los servicios de voz paquetizados hoy en día, gracias a su elevado nivel de control sobre la calidad de servicio (QoS). ATM permite a los administradores de la red configurar conexiones que cumplen requisitos específicos tales como garantías de ancho de banda o retardos mínimos establecidos para una aplicación determinada. El estándar predominante para Voz ATM es

el "Trunking ATM", que proporciona compresión de voz, supresión de silencios y demodulación de fax.

Sin embargo, ATM es aún una tecnología muy compleja, y el equipamiento de acceso ha sido siempre significativamente más caro que las soluciones disponibles para Frame Relay o IP. Al igual que Frame Relay, ATM ha sido diseñada e implantada principalmente como tecnología de troncal y no es muy probable que se llegue a grandes despliegues en entornos de grupo de trabajo LAN. Y sin ATM en la LAN, las soluciones VoATM son difíciles de integrar completamente en los entornos corporativos.

3.2.2.3.-Voz sobre IP

Como vimos en capítulos anteriores, Voz sobre IP (VoIP, Voice over IP) es una tecnología que permite la transmisión de la voz a través de redes IP en forma de paquetes de datos.

La Telefonía IP es una aplicación inmediata de esta tecnología, de forma que permita la realización de llamadas telefónicas ordinarias sobre redes IP u otras redes de paquetes utilizando un PC, gateways y teléfonos estándares. En general, servicios de comunicación son transportadas vía redes IP, Internet normalmente, en lugar de ser transportados vía la red telefónica convencional.

3.2.3.-Codificadores.

Estos códec son una parte importante dentro de nuestro sistema de telefonía puesto que son los encargados de convertir la señal analógica a digital y además de la conversión tiene como característica la de comprimir la secuencia de datos y la cancelación del eco.

Una parte sumamente importante es el cálculo del ancho de banda para saber así el códec más adecuado para poder usar la telefonía IP sin tener que sacrificar nitidez en la comunicación. Además se tiene sumar a este cálculo la estimación de llamadas simultáneas que se necesitaran para poder medir el ancho de banda.

Codec's mas populares

G.711 (PCM): banda ethernet 95.2 kbps

G.723.1A (ACELP): banda ethernet 26.1 kbps

G.723.1A (MP-MLQ): banda ethernet 27.2 kbps

G.726 (ADPCM): banda ethernet 63.2 kbps

G.728 (LD-CELP): banda ethernet 78.4 kbps

G.729a (CS-CELP): banda ethernet 39.2 kbps

AMR-WB/G.722.2 (ACELP): banda ethernet 38.0 kbps

Tabla 3. 6.-Codec's populares.

3.3.-Componentes

3.2.1.- Terminales Voip (Teléfonos IP)

Es un dispositivo que permite realizar una comunicación utilizando una red IP ya sea mediante red de área local o a través de Internet. En telefonía IP nos referimos a un *terminal IP* ya que son los principales dispositivos utilizados para realizar una comunicación de paquetes de datos en los que se transporta voz o vídeo.



Figura 3. 24.-Teléfono IP.

- Un *terminal IP* suele ser un dispositivo hardware con forma de teléfono, aunque con la diferencia de que utiliza una conexión de red de datos, en lugar de una conexión de red telefónica.
- Suelen tener más opciones y ventajas que un teléfono convencional. Al ser un sistema completamente digital y programable, suelen tener teclas especiales perfectamente configurables mediante un sistema de administración que puede ser accedido mediante web o mediante telnet.

- Algunos incluyen cámara de vídeo para poder realizar videoconferencias.
- Disponen de una dirección IP a la que poder acceder y mediante la que se puede configurar como si fuese un ordenador más. Por lo que, al considerarse un sistema más dentro de la red, suelen aplicárseles las características típicas de grandes redes.

3.2.2.- Terminales VoIP (Softphone)

Un **Softphone** es un software que hace una simulación de teléfono convencional por computadora. Es decir, permite usar la computadora para hacer llamadas a otros softphones. Un Softphone es parte de un entorno Voz sobre IP y puede estar basado en el estandard SIP/H.323.

Algunos softphones están implementados completamente en software, que se comunica con las PBX a través de la (LAN) Red de Área Local.



Figura 3. 25.-Softphone.

3.4.-Beneficios de implantar VoIP.

Los beneficios de implantar VoIP en Luz y Fuerza se dividirían en diversos ámbitos, por un lado el claro ahorro telefónico, por otro lado el incremento de productividad de los empleados al disponer de ciertas herramientas en su teléfono. Se reduciría de manera muy importante el gasto en llamadas internas. Además, VoIP ofrece la posibilidad de configurar el sistema y parametrizar cualquier cambio de nuevas extensiones.

CAPÍTULO 4

INSTALACIÓN Y CONFIGURACIÓN DE ASTERISK (SERVIDOR)

4.1.- Asterisk

Asterisk es un PBX por software, el cual puede servir en Linux, BSD, MacOSX o Windows, este permite construir aplicaciones de comunicación tan complejas o avanzadas como se desee, sin incurrir en altos costos y con más flexibilidad que en cualquier sistema de telefonía.

La definición tradicional sería “Transmisión de señales de voz codificadas a través de una red de datos IP”.

Como se mencionó en el capítulo dos Asterisk es un software que se encarga de transformar una computadora (PC o servidor) en una centralita telefónica (software) de código abierto. La cual permite comunicar vía voz, a dos o más terminales ip para así sostener una comunicación entre las terminales. Asterisk está desarrollado para ser flexible.

4.1.1.- ¿Cómo Funciona?

Asterisk funciona sin la necesidad de hardware alguno para su conectividad con la PSTN, siendo esta la gran ventaja para este proyecto el cual está basado en Voz sobre IP. El software nos entrega conectividad en tiempo real para redes IP sin tener que reestructurar dicha red.

Se pueden conectar varios tipos de teléfono: Análogos (convencionales), IP o virtuales. Los teléfonos análogos son los teléfonos más económicos que hay, pero requieren de un adaptador de voz (ATA) para conectarse a la central. Los teléfonos IP son teléfonos digitales que se conectan directamente a un punto de red. Los teléfonos USB, se conectan a un PC para funcionar, y los teléfonos virtuales son un software en un PC auxiliados con un micrófono y unos audífonos y solo requiere de una tarjeta de sonido.

4.1.2.- Componentes de Asterisk.

Asterisk contiene en su núcleo cuatro partes:

- **PBX SWITCHING:** Es un sistema de conmutación de intercambio, conectando llamadas entre varios usuarios y tareas automatizadas. La base de conmutación conecta a los usuarios llegando a varios software y hardware de interface.

- **LANZADOR DE APLICACIONES:** Lanza aplicaciones que mejoran servicios para usos tales como, voicemail, file playback y lista de directorio.
- **TRADUCTOR DE CODECS:** usa módulos de Códecs para codificar y decodificar varios formatos de comprensión de audio usado en la industria de la telefonía. Un gran número de Códecs están disponibles para satisfacer necesidades y llegar al mejor balance entre la calidad del audio.
- **ORGANIZADOR Y MANEJADOR:** Maneja la organización de tareas de bajo nivel y sistemas de manejo para un óptimo performance bajo cualquier condición de carga.

Además Asterisk contiene cuatro APIs:

- **CANAL API:** El canal API maneja el tipo de conexión al cual el cliente está llegando, sea una conexión VoIP, ISDN, PSTN, o algún otro tipo de tecnología. Módulos dinámicos son cargados para manejar los detalles más bajos de estas conexiones.
- **APLICACIÓN API:** Esta aplicación permite a varios módulos de tareas cumplir varias funciones , conferencias, lista de directorios, voice mail en la línea de transmisión de datos , y cualquier otra tarea la cual PBX sea capaz de cumplir ahora o en el futuro son manejados por estos módulos.
- **TRADUCTOR DEL CODEC API:** Cargar módulos codecs para apoyar varios tipos de audio, codificando y decodificando formatos.
- **FORMATO DE ARCHIVO API:** Maneja la lectura y escritura de varios formatos de archivos para el almacenaje de datos en el sistema de archivos.

Cargar todo en módulos le permite a Asterisk ser un sistema flexible, y así diseñar la mejor y más satisfactoria trayectoria para los usuarios en el sistema PBX y también modificar el DialPlan para satisfacer las necesidades de la comunicación.

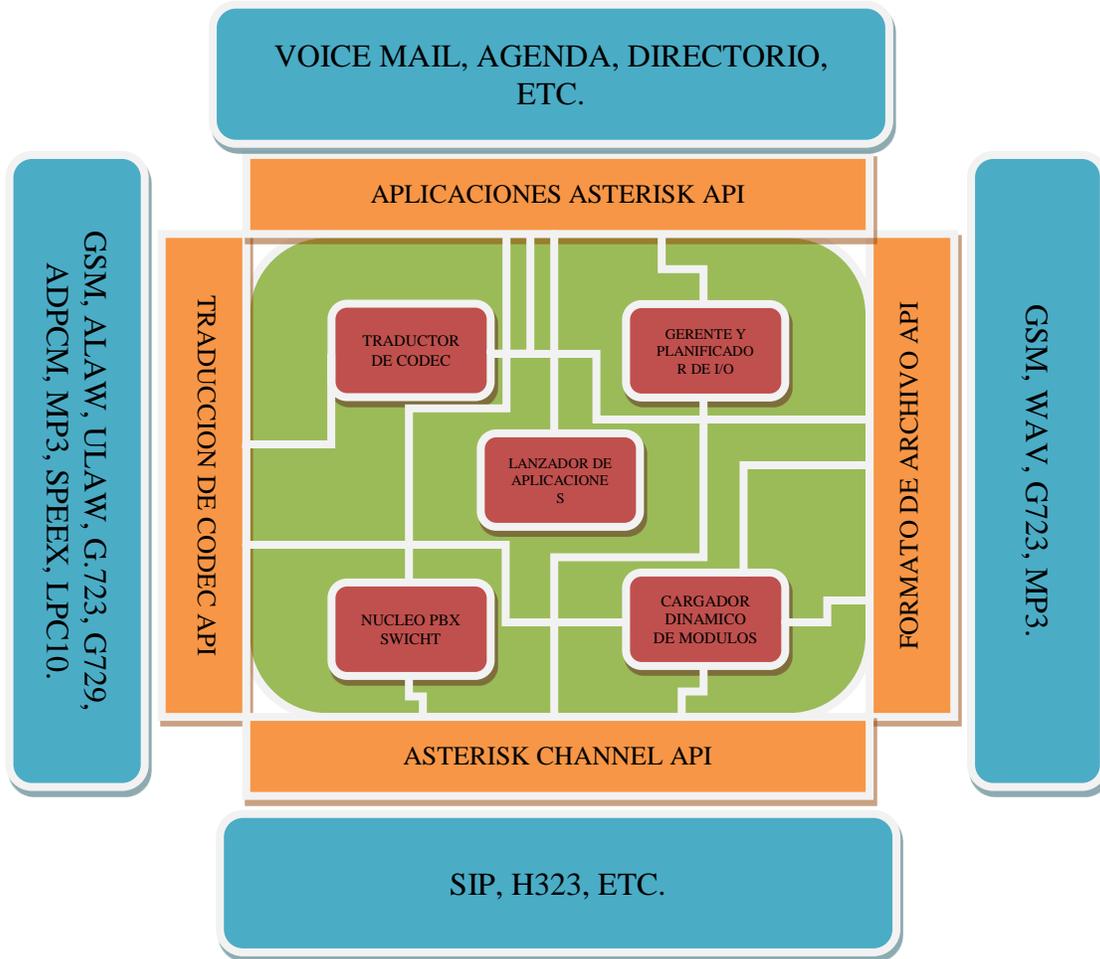


Figura 4.1.- Esquema de los componentes Asterisk

4.1.3.- Software necesario

El software como ya se comentó en esta tesis, y por motivos de economía y ahorro de recursos del proyecto se tomó la decisión de implementar el sistema bajo un ambiente libre de licencias y así brindar un servicio mejor, a menor costo, tanto del sistema operativo como los programas necesarios para que Asterisk funcione correctamente.

La instalación de Asterisk requiere de 4 cosas fundamentales para la correcta funcionalidad del servicio y dependerá en gran parte de las tarjetas que se vayan a utilizar en el servidor.

Las cuatro partes que componen Asterisk son:

- **Libpri**
Estas son las librerías necesarias para poder operar con las tarjetas de comunicaciones que utilizan la tecnología Zap.
- **Zaptel o dadhi**
Son las APIs de los drivers de las tarjetas que sirven para implementar la plataforma Asterisk para la voz ip. En otras palabras son los drivers de las tarjetas de comunicación. Desde el año 2006 se determinó el cambio de nombre de Zaptel a Dahdi(Digium Asterisk Hardware Device Interface).
- **Asterisk**
Este es el paquete principal encargado de mover todo lo que significa Asterisk, dentro de este paquete se encuentran los canales que podemos utilizar, las aplicaciones, funciones, recursos, etc.
- **Asterisk-addons**
Paquetes opcionales que sólo funcionan anexados a Asterisk y que sirven para incrementar o complementar sus funcionalidades.

4.1.3.1.-Sistema Operativo.

El Sistema Operativo que elegimos para la implementación de Asterisk es Linux por sus prestaciones actuales podemos decir que es la solución más adecuada para los objetivos de este proyecto.

Linux en pocas palabras nos brindara mayor efectividad en cuanto a velocidad, economía y aprovechamiento de los recursos del servidor y de la red de datos.

Algo que no debemos olvidar es que Linux es un sistema bajo licencia GPL, y es por esta razón que fue la mejor opción para la implementación de Asterisk en este proyecto, además de que Asterisk fue diseñado en primer lugar para trabajar con Linux y hasta el momento es el Sistema Operativo que mejor soporta a Asterisk.

Debian (http://www.debian.org)	Fedora (http://fedora.redhat.com)
Ubuntu (http://www.ubuntu.com)	RedHat (http://www.redhat.com)
CentOS (http://www.centos.org)	SuseOS (http://www.suse.com)

Tabla 4.1.-Distribuciones.

4.1.3.2.- Paquetes (Software necesario)

Para empezar con la instalación de Asterisk se necesitan varios paquetes para la correcta funcionalidad de Asterisk.

1	gcc : Compilador de C y sus dependencias(se sugiere la versión 3.0 o superior).
2	g++: Compilador de C++.
3	make: Automatizador para la compilación rápida y fácil.
4	bison (necesario para los archivos de configuración).
5	ncurses (necesario para la consola o CLI).
6	openssl y openssl-dev o libssl-dev (necesario para las librerías criptográficas).

Tabla 4.2.- Paquetes necesarios.

4.1.3.3.- Librerías

Las librerías sirven para compilar Asterisk en nuestro Sistema Operativo Linux. También necesitamos hacer un análisis de hardware, para saber si es necesario instalar más librerías, aunque en este proyecto el hardware es mínimo, por lo tanto las librerías básicas nos van a servir para que nuestro sistema PBX funcione correctamente.

Librerías necesarias:

1	libssl-dev: Librería necesaria para gestionar tramas cifradas.
2	zlib1g-dev: Librería para comprimir y descomprimir datos mediante el algoritmo zlib.
3	libncurses5-dev: Librería para utilizar el modo pseudo-gráfico en la consola de texto.
4	libnewt-dev: Librería para mostrar un interfaz de color en la consola de texto.
5	kernel headers o kernel sources: Los archivos <i>includes</i> o bien el código fuente del kernel de Linux que estemos utilizando.
6	Build-essential.- paquetes necesarios para compilar.

Tabla 4.3.- Librerías necesarias.

La versión depende mucho de la compatibilidad con el paquete principal (Asterisk) por lo que ocuparemos versiones compatibles con Asterisk 1.4.X, no obstante las versiones se deben de actualizar cada mes aproximadamente, puesto que Digium, la empresa encargada de dar soporte a la aplicación lanza cada mes avisos y/o parches de programas o librerías a fin de lograr mucha más seguridad y disminución de colapsos en los procesos de Asterisk.

4.2.- Instalación del Sistema Operativo

Como se dijo, usaremos Debian como Sistema Operativo, por tener el mejor soporte por parte de los usuarios y el kernel será el 2.6 para una mejor compatibilidad con Asterisk, el sistema se instalara desde el inicio, pensando siempre en la compatibilidad con Asterisk.

Para descargar la versión estable de Debian accedamos a la página <http://www.debian.org/> y descargamos la versión netinst que pesa aproximadamente 180Mb, la cual contiene un sistema base y algunos programas básicos de Debian para empezar a utilizar el Sistema Operativo, y así comenzar con nuestra instalación.



Figura 4.2.- Instalación de Debian

En el momento de insertar el CD aparecerá la siguiente imagen deberemos seleccionar install para instalar Debian en modo texto.

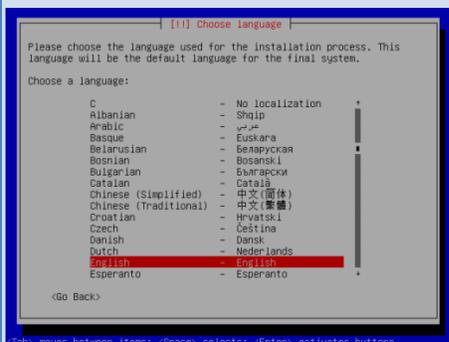
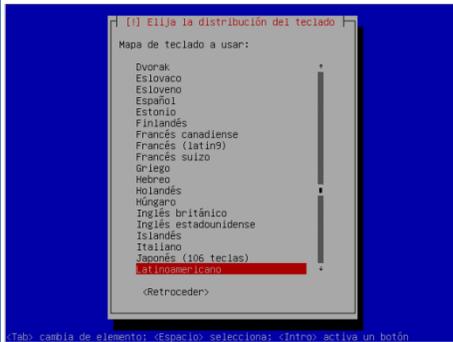


Figura 4.3.- Idioma instalación

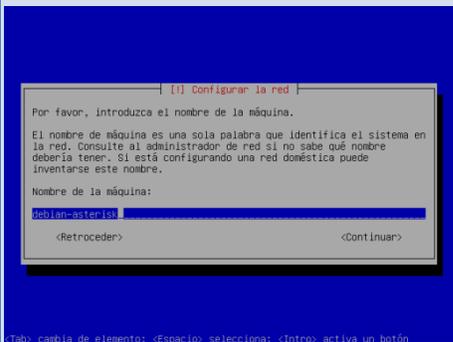
Una vez empezada la instalación el sistema instalador nos pide seleccionar el idioma, seguida de la ubicación, deberemos de seleccionar Spanish seguido de México.



Tab) cambia de elemento: <Espacio> selecciona: <Intro> activa un botón

Figura 4.4.- Configuración teclado

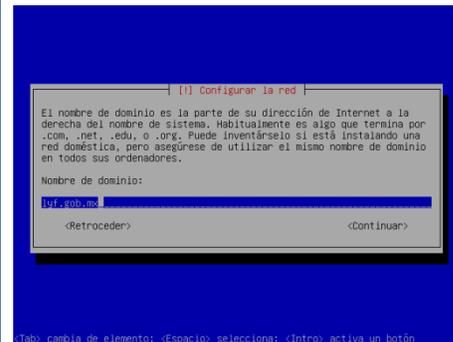
Linux es un sistema sensible al teclado, lo que significa que distingue mayúsculas y minúsculas por lo que la distribución del teclado es esencial por lo que seleccionamos Latinoamérica.



Tab) cambia de elemento: <Espacio> selecciona: <Intro> activa un botón

Figura 4.5.- Configuración Red

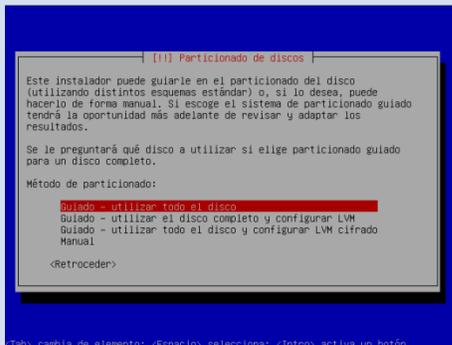
Ponemos después el nombre de la máquina para que el equipo pueda identificarse en la red.



Tab) cambia de elemento: <Espacio> selecciona: <Intro> activa un botón

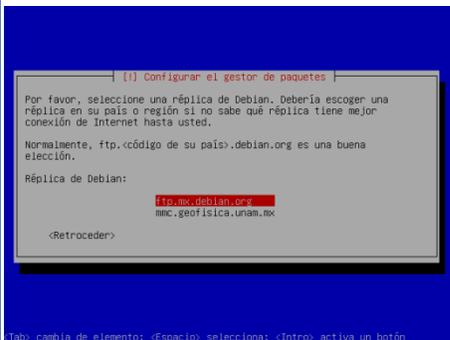
Figura 4.6.-Configuración dominio

También debemos poner el dominio de la máquina para poder acceder a la red de la empresa el cual es lyf.gob.mx.



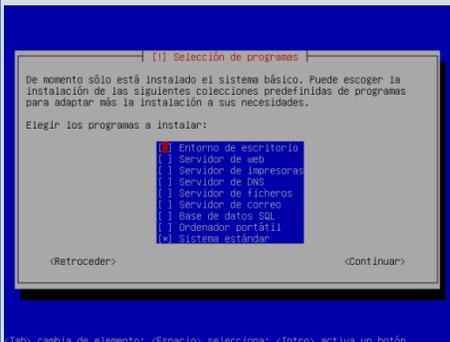
Tablo cambia de elemento: [Espacio] selección: [Intro] activa un botón
Figura 4.7.- Configuración Reloj

Nos pedirá la zona horaria para configurar el reloj y posteriormente el particionamiento de los discos se debe elegir utilizar todo el disco, aunque lo mas recomendable es hacer la partición manual para configurar nuestro sistema separándolo en por lo menos 4 particiones. Finalizar el particionado y aceptar para escribir los cambios en el disco.



Tablo cambia de elemento: [Espacio] selección: [Intro] activa un botón
Figura 4.8.-Configuración Usuarios

Configurar usuarios y sus contraseñas, tanto del usuario root como de un usuario para el sistema. Y después se debe de configurar el gestor de paquetes por lo que debemos seleccionar México, y después fpt.mx.debian.org y no poner ningún proxy.



Tablo cambia de elemento: [Espacio] selección: [Intro] activa un botón
Figura 4.9.- Paquetería.

Una vez terminados los procedimientos anteriores el sistema descargara e instalara los archivos necesarios del repositorio que seleccionamos, y nos preguntara que queremos instalar, a lo cual seleccionaremos [*]sistema estándar, las demás opciones no las necesitamos.

	<p>El sistema descargara los ficheros necesarios para configurar el sistema operativo conforme a las características que elegimos.</p> <p>Pedirá que si deseamos instalar el cargador GRUP en el registro principal de arranque, le decimos que si, ya que el servidor tendrá únicamente Linux como Sistema Operativo.</p>
--	--

Figura 4.10.- Instalación GRUP

Tabla 4.4.- Instalación de Debian.

4.3.- Instalación de Asterisk

Para iniciar con la instalación de Asterisk no solamente es necesario instalar el sistema base de Debian, sino que también se necesita el grupo de paquetes antes mencionados, así como también sus dependencias y librerías para su correcto funcionamiento, más adelante se mostrara como instalar cada una de las dependencias para la correcta instalación de Asterisk.

Instalación de gcc más sus dependencias, para instalar gcc solo con escribir apt-get install gcc se encargara de instalar la versión más reciente del paquete, así como las

```

debian-asterisk:~# apt-get install gcc
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes extras:
  binutils cpp cpp-4.3 gcc-4.3 libc6-dev libgmp3c2 libgomp1 libmpfr1ldbl
  linux-libc-dev
Paquetes sugeridos:
  binutils-doc cpp-doc gcc-4.3-locales gcc-multilib make manpages-dev autoconf
  automake1.9 libtool flex bison gdb gcc-doc gcc-4.3-multilib
  libmudflap0-4.3-dev gcc-4.3-doc libgcc1-dbg libgomp1-dbg libmudflap0-dbg
  glibc-doc
Se instalarán los siguientes paquetes NUEVOS:
  binutils cpp cpp-4.3 gcc gcc-4.3 libc6-dev libgmp3c2 libgomp1 libmpfr1ldbl
  linux-libc-dev
0 actualizados, 10 se instalarán, 0 para eliminar y 0 no actualizados.
Necesito descargar 13.2MB de archivos.
Se utilizarán 38.6MB de espacio de disco adicional después de esta operación.
¿Desea continuar [S/n]? S_

```

dependencias, y al final de la muestra de las dependencias solo basta con escribir una S para aceptar.

Los paquetes que son complementarios de gcc son:

Binutils, cpp, cpp-4.3, libgmp3c2, libmpfr1ldbl, linu-libc-dev.

Figura 4.11.- Instalación de gcc.

```

debian-asterisk:~# apt-get install g++
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes extras:
  g++-4.3 libstdc++6-4.3-dev
Paquetes sugeridos:
  g++-multilib g++-4.3-multilib gcc-4.3-doc libstdc++6-4.3-dbg
  libstdc++6-4.3-doc
Se instalarán los siguientes paquetes NUEVOS:
  g++ g++-4.3 libstdc++6-4.3-dev
0 actualizados, 3 se instalarán, 0 para eliminar y 0 no actualizados.
Necesito descargar 4804kB de archivos.
Se utilizarán 18.3MB de espacio de disco adicional después de esta operación.
¿Desea continuar [S/n]? _

```

Instalar g++ y sus dependencias son:
 g++-4.3
 Libstdc++6-4.3-dev

Figura 4.12.- Instalación de G++

```

Desempaquetando g++ (de ../g++_4%3a4.3.2-2_i386.deb) ...
Procesando disparadores para man-db ...
Configurando g++-4.3 (4.3.2-1.1) ...
Configurando libstdc++6-4.3-dev (4.3.2-1.1) ...
Configurando g++ (4:4.3.2-2) ...
debian-asterisk:~# apt-get install make
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Paquetes sugeridos:
  make-doc
Se instalarán los siguientes paquetes NUEVOS:
  make
0 actualizados, 1 se instalarán, 0 para eliminar y 0 no actualizados.
Necesito descargar 382kB de archivos.
Se utilizarán 991kB de espacio de disco adicional después de esta operación.
Des:1 http://mmc.geofisica.unam.mx lenny/main make 3.81-5 [382kB]
Descargados 382kB en 2s (141kB/s)
Seleccionando el paquete make previamente no seleccionado.
(Leyendo la base de datos ...
22681 ficheros y directorios instalados actualmente.)
Desempaquetando make (de ../archives/make_3.81-5_i386.deb) ...
Procesando disparadores para man-db ...
Configurando make (3.81-5) ...
debian-asterisk:~# _

```

Make: Automatizador para la compilación rápida y fácil.

Figura 4.13.- Instalación de Make

```

debian-asterisk:~# apt-get install build-essential linux-headers-$(uname -r) libncurses-dev libnewt-dev libssl-dev zlib1g
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Nota, seleccionando libncurses5-dev en lugar de libncurses-dev
zlib1g ya está en su versión más reciente.
Se instalarán los siguientes paquetes extras:
  bzip2 cpp-4.1 dpkg-dev gcc-4.1 gcc-4.1-base libncurses5-dev libpng12-0
  libpng12-dev libslang2-dev libtimedate-perl linux-headers-2.6.26-2-common
  linux-kbuild-2.6.26 zlib1g-dev
Paquetes sugeridos:
  bzip2-doc gcc-4.1-localles debian-keyring gcc-4.1-multilib gcc-4.1-doc
  libmudflap0-dev
Se instalarán los siguientes paquetes NUEVOS:
  build-essential bzip2 cpp-4.1 dpkg-dev gcc-4.1 gcc-4.1-base libncurses5-dev
  libnewt-dev libpng12-0 libpng12-dev libslang2-dev libssl-dev
  libtimedate-perl linux-headers-2.6.26-686 linux-headers-2.6.26-2-common
  linux-kbuild-2.6.26 zlib1g-dev
0 actualizados, 17 se instalarán, 0 para eliminar y 0 no actualizados.
Necesito descargar 12.9MB de archivos.
Se utilizarán 58.9MB de espacio de disco adicional después de esta operación.
¿Desea continuar [S/n]? _

```

Instalar el paquete de linux-headers de la versión del kernel que en este caso es el 2.6.26, este paquete se podrá instalar con el siguiente comando:

Figura 4.14.- Instalación de Linux- Headers

```
apt-get install linux-headers-$(uname -r)
```

Con el comando anterior la versión del paquete que se debe instalar será exactamente la misma que se tiene instalada.

En este momento nuestro sistema ya tiene todo lo necesario para poder instalar y compilar Asterisk. Procederemos a descargar los paquetes de Asterisk desde la página <http://www.asterisk.org/downloads>, el primer paso será tener los paquetes y posicionarnos en `/usr/src/`.

Para descargarlos desde consola se tendrá que utilizar el siguiente comando:

```
wget http://downloads.asterisk.org/pub/telephony/asterisk/asterisk-1.4.28.tar.gz
```

```
debian-asterisk:~# wget http://downloads.asterisk.org/pub/telephony/asterisk/ast
erisk-1.4.28.tar.gz
--2009-12-30 16:50:27-- http://downloads.asterisk.org/pub/telephony/asterisk/as
terisk-1.4.28.tar.gz
Resolviendo downloads.asterisk.org... 76.164.171.233
Connecting to downloads.asterisk.org|76.164.171.233|:80... conectado.
Petición HTTP enviada, esperando respuesta... 200 OK
Longitud: 21451566 (20M) [application/x-gzip]
Saving to: 'asterisk-1.4.28.tar.gz'
52% [=====] 11,320,385 197K/s eta 51s
```

En la figura 4.15 se ve el progreso de la descarga, así como también el servidor al que estamos conectados, siendo así es correcto.

Figura 4.15.- Descarga de Fuentes (Asterisk)

Después de descargar el archivo mediante `http` se tiene que desempaquetar utilizando el comando:

```
Asterisk # tar -xvzf.
```

Una vez que tengamos el directorio tendremos que entrar y configurar el proceso de compilación para eso usaremos el comando: `/configure` una vez dentro del directorio, para asegurarnos de estar en el directorio correcto podemos ayudarnos del comando `pwd` el cual despliega el directorio en el que estamos posicionados.

Para corroborar que el proceso de Asterisk está en ejecución, solamente bastara teclear en consola con permisos de root, el siguiente comando:

```
ps aux | grep asterisk
```

Este comando muestra los procesos en ejecución y filtrados todos los que tengan el nombre de Asterisk.

Ahora ya tenemos un sistema Asterisk instalado en nuestro servidor, aunque viene con opciones por defecto. De aquí en adelante se podrá configurar nuestro sistema de centralita VoIP, aunque primero instalaremos las demás utilerías para que nuestro sistema quede completamente instalado y después continuar con el proceso de configuración como lo dice el temario.

4.4.-Instalación de Asterisk-gui

Este software nos va a permitir la administración de Asterisk vía web, de una manera un poco más sencilla sin entrar a consola. Aunque sea un poco más fácil, la manera adecuada de configurar Asterisk es por medio de consola y sin tener ningún proceso innecesario en el servidor.

Para su instalación se deberá conseguir la versión más reciente que en este momento es la asterisk-gui-2.0.4.tar.gz por lo tanto tendremos que situarnos de nuevo en el directorio /usr/src/ para después ocupar nuevamente el comando:

```
wget http://downloads.asterisk.org/pub/telephony/asterisk-gui/releases/asterisk-gui-2.0.4.tar.gz
```

Una vez terminada la descarga debemos extraerlo como lo hicimos con el paquete de Asterisk con el comando:

```
tar -xvzf asterisk-gui-2.0.4.tar.gz
```

Entraremos en el directorio creado para configurar, compilar e instalar, por lo tanto lo haremos con los mismos pasos que en la instalación de Asterisk así que el comando a utilizar será:

```
Debian-Asterisk:/usr/src/asterisk-gui-2.0.4 $ ./configure
```

Compilamos con “make”:

```
Debian-Asterisk:/usr/src/asterisk-gui-2.0.4 $ make
```

Instalación con make install:

```
Debian-Asterisk:/usr/src/asterisk-gui-2.0.4 $ make install
```

Una vez terminado este proceso ya está instalado Asterisk-gui, pero se necesita activar ciertas cosas antes de poder utilizarlo.

Tengo que comentar que Asterisk-gui comunica con Asterisk mediante AMI (Asterisk Manager Interface) y la gran ventaja es que no necesita ningún servidor web puesto que el sistema trae el suyo por defecto.

Primero tendremos que ir al directorio /etc/asterisk para ahí editar el archivo de configuración de AMI el cual se llama manager.conf, otra vez les recuerdo que este proceso se ejecuta como root para poder modificar el archivo, de lo contrario nos dará un error al intentar salvarlo.

Este archivo de configuración contiene líneas definidas y muchas otras líneas de ayuda, las líneas que empiecen con “;” son las líneas que no tienen efecto para Asterisk, al eliminar el punto y coma activamos el parámetro seguido.

Las líneas básicas que debe contener el archivo manager.conf son las siguientes:

```
;inicio del archivo
[general]
displayssystemname = yes
enabled = yes
webenabled = yes
port = 5038
bindaddr = 10.10.106.100
```

Ahora, en este archivo básicamente lo que hicimos fue habilitar el Asterisk AMI con la línea de enable= yes y con webenabled=yes, definimos el puerto que utiliza por defecto le dejamos el 5038 y le indicamos desde que dirección se tendrá acceso a AMI, que en este caso fue la misma dirección del servidor aunque cabe destacar que Asterisk-gui se pude instalar en un servidor diferente y así los procesos que genera este paquete dejárselos a un servidor distinto, para no interferir con los procesos que requiere Asterisk y ahorrar procesador para que Asterisk no tenga ningún obstáculo que pueda causar fallos en los

procesos internos de Asterisk, en este caso ocuparemos el mismo servidor con la dirección 10.10.106.100 o podemos dejar (127.0.0.1).

Al final del archivo definiremos al usuario [administrador] de Asterisk, y le asignamos una contraseña en este caso "123456". Además definimos los roles de usuario que tendrá el control total. Y quedaría de la siguiente manera:

```
[administrador]
secret = 123456
read = system,call,log,verbose,command,agent,user,config
write = system,call,log,verbose,command,agent,user,config
;fin del archivo
```

Al término de la definición de usuario tendremos que activar el servidor http de Asterisk, esto lo hacemos editando el archivo http.conf ubicado en /etc/asterisk.

```
[general]
enabled = yes
enablestatic = yes
bindaddr = 0.0.0.0
bindport = 8088
prefix = asterisk
```

Ejemplos de configuración (depende de dónde nos conectemos):

127.0.0.1 si es desde la propia maquina
192.168.0.0 desde la red local
0.0.0.0 para cualquier sistema desde Internet.

El parámetro bindaddress con valores 0.0.0.0 es la mejor opción, puesto que podemos acceder desde cualquier dirección IP de la red de la empresa.

Para continuar después de haber editado estos archivos, es necesario reiniciar Asterisk para que los cambios tomen efecto, por lo tanto es necesario detener Asterisk con el comando:

```
Debian-Asterisk: /etc/asterisk $ killall asterisk
```

Y luego volver a arrancarlo con la orden:

```
Debian-Asterisk:/etc/asterisk $ asterisk
```

Para asegurarnos de que Asterisk este corriendo en el servidor debemos poner `netstat -atupn | grep asterisk` en consola, para visualizar que conexiones tienen el proceso

(Figura 4.19) Asterisk, las que nos interesan son dos conexiones en particular la conexión AMI que se aprecia por tener puerto 5038 y el Asterisk-gui en el puerto 8088.

```
debian-asterisk:~# netstat -atupn | grep asterisk
tcp        0      0 127.0.0.1:5038      0.0.0.0:*          LISTEN
2166/asterisk
tcp        0      0 0.0.0.0:2000       0.0.0.0:*          LISTEN
2166/asterisk
tcp        0      0 0.0.0.0:8088       0.0.0.0:*          LISTEN
2166/asterisk
udp        0      0 0.0.0.0:2727       0.0.0.0:*
2166/asterisk
udp        0      0 0.0.0.0:4520       0.0.0.0:*
2166/asterisk
udp        0      0 0.0.0.0:5060       0.0.0.0:*
2166/asterisk
udp        0      0 0.0.0.0:4569       0.0.0.0:*
2166/asterisk
debian-asterisk:~# _
```

Figura 4.19.-Vista de Procesos

Una vez concluida la configuración podemos verificar con un comando dentro del directorio `/usr/src/asterisk-gui-2.0.4` no olvidemos que tenemos que ejecutarlo como root.

```
Debian-Asterisk:/usr/src/asterisk-gui-2.0.4 $ make checkconfig
```

Este comando realiza una serie de comprobaciones y si no encuentra ningún error de configuración nos dará como resultado la liga donde tendremos que entrar para la administración web.

La manera de revisar que todo ha salido bien en esta instalación es, poner en un explorador web la liga que se muestra al concluir el `make checkconfig`.

4.5.- Directorios de instalación de Asterisk

Los archivos de configuración de Asterisk se encuentran en el directorio `/etc/asterisk` los cuales tienen una extensión “.conf”.

Asterisk configura los directorios de instalación y configuración. El archivo encargado de realizar esta misión es: `/etc/asterisk/ asterisk.conf`.

```

GNU nano 2.0.7          Fichero: asterisk.conf
[directories]
astetcdir => /etc/asterisk
astmoddir => /usr/lib/asterisk/modules
astvarlibdir => /var/lib/asterisk
astdatadir => /var/lib/asterisk
astagidir => /var/lib/asterisk/agi-bin
astspooldir => /var/spool/asterisk
astrundir => /var/run
astlogdir => /var/log/asterisk

[options]
languageprefix = yes ; Use the new sound prefix path syntax
;verbose = 3
;debug = 3
;alwaysfork = yes ; same as -F at startup
;nofork = yes ; same as -f at startup
;quiet = yes ; same as -q at startup
;timestamp = yes ; same as -T at startup
;execincludes = yes ; support #exec in config files
;console = yes ; Run as console (same as -c at startup)

^G Uer ayuda  ^O Guardar  ^R Leer Fich ^Y Pág Ant  ^K CortarTxt ^C Pos actual
^X Salir     ^J Justificar ^W Buscar   ^U Pág Sig  ^J PegarTxt  ^I Ortografía

```

Figura 4.20.- Directorios de Instalación.

En este archivo se encuentran las rutas de los archivos que intervienen en la funcionalidad de Asterisk, podremos modificarlo a nuestras necesidades, aunque viene con la configuración por default. Lo que se sugiere es no modificarlo pero durante la instalación del sistema operativo separar y montar las rutas /etc y /var en otra partición diferente a la del directorio raíz.

Directorio	Descripción
/etc/asterisk/	Aquí residen los archivos de configuración de Asterisk
/usr/lib/asterisk/modules/	Este directorio contiene los módulos de Asterisk
/usr/sbin/	Aquí reside el binario de Asterisk
/var/log/asterisk/	Contiene los logs de Asterisk
/var/lib/asterisk/agi-bin/	Directorio donde residen los <i>scripts</i> AGI
/var/lib/asterisk/mohmp3	Carpeta que contiene archivos para música en espera
/var/lib/asterisk/sounds	Sonidos que Asterisk utiliza
/var/spool/asterisk/	Directorio donde Asterisk guarda archivos que genera producto de su funcionamiento como <i>voicemails</i> y grabaciones de llamadas
/var/run/	Archivos con información de PIDs
/var/log/asterisk/	Aquí residen los archivos de <i>log</i> de Asterisk como el /var/log/asterisk/full o el <i>log</i> de texto de CDRs

Tabla 4.5.- Directorios de instalación.

Asterisk se las ha arreglado para extender sus funcionalidades y se enfoca a la idea de módulos para separar procesos, y así tener una mejor funcionalidad. Estos módulos se encuentran en la carpeta /usr/lib/asterisk/modules/ y tienen la extensión (.so).

Los módulos se pueden administrar a través del archivo de configuración modules.conf. En él se le puede decir a Asterisk que “cargue” ó “no cargue” un módulo específico con la directiva “load” y “noload”.

Este es un ejemplo de cómo cargan los módulos.

```
noload => chan_modem_bestdata.so
noload => chan_modem_i4l.so
noload => app_trunkisavail.so
load => format_wav.so
load => format_pcm.so
load => format_au.so
```

4.6.-Canal

Un canal es el equivalente a una línea telefónica en la forma de un circuito de voz digital. Este generalmente consiste de una señal analógica en un sistema POTS o alguna combinación de CODEC y protocolos de señalización (SIP, IAX).

Algunos canales para la configuración de Asterisk son:

CANAL	DESCRIPCIÓN
adtranvofr.conf	Configurar los canales de voz sobre Frame Relay (al estilo de Adtran)
agents.conf	Configurar los canales de agente
h323.conf	Configurar los canales H323
iax.conf	Configurar los canales IAX
mgcp.conf	Configurar los canales MGCP
phone.conf	Configurar los canales de teléfono (dispositivos de telefonía para Linux)
sip.conf	Configurar los canales SIP
sip_notify.conf	Configurar los mensajes SIP NOTIFY
skinny.conf	Configurar los canales Skinny (Cisco SCCP)
vpb.conf	Configurar los canales vpb (tarjetas de Voicetronix)
zapata.conf	Configurar los canales Zap(tarjetas de Digium)

Tabla 4.6.- Archivos de configuración para los Canales

Para este proyecto solamente se ocupara el canal SIP por lo ya anteriormente comentado en el capítulo 3, por ser la mejor opción para los requerimiento y necesidades de nuestro sistema.¹⁹

4.7.-Gramática de Asterisk

Asterisk requiere una sintaxis muy especial para sus archivos de configuración, algunas sintaxis son parecidas a la de los lenguajes de programación, a continuación se muestra el significado de los signos más comunes los archivos de configuración de Asterisk.

Nota 1: "El canal SIP y el canal IAX2 se configuran de la misma manera".

Signos	Ejemplo
El signo “;” significa comentarios	;este es un comentario
El “=” sirve para declarar valores u objetos	llave= variable;
El “=>” sirve para declarar valores u objetos	objeto=>valor;

Tabla 4.7.- Signos

Gramática	Creación de los objetos	Configuración en el archivo	Ejemplo
Grupo simple	Todos se encuentran en la misma línea	Extension.conf	Exten=>203,1,Dial(SIP/203)
Herencia de opciones	Primero se definen las opciones, para después hacer que los objetos las hereden	dahdi-channels.conf	[channels] context=default signalling=fxs_ks group=1 channel=>1
Entidad compleja	Cada entidad recibe un contexto	Sip.conf Iax.conf	[cisco] type=friend secret=mysecret host=10.10.106.101 context=trusted [xlite] type=friend secret=xlite host=dynamic

Tabla 4.8.- Tipos de Gramáticas.

4.7.1.-Gramática de grupo simple

En este caso los objetos son declarados con todas las opciones dentro de la misma línea como se describe a continuación.

```
[sección]
objeto1 => op1, op2, op3
objeto2 => op1b, op2b, op3b
```

Los archivos como extensions.conf, meetme.conf, voicemail.conf siguen este formato.

4.7.2.-Gramática de Herencia de Opciones

```
[sección]
op1=bas           ;línea 1
op2=adv           ;línea 2
objeto=>1         ;línea 3
op1=int           ;línea 4
objeto=>2         ;línea 5
```

Las primeras dos líneas configuran las opciones op1 y op2 que son bas y adv, una vez inicializado el objeto 1 en la línea 3 este toma las opciones anteriores a él, es decir op1= bas y op2=adv. En la línea 4 la op1 cambia su valor a int, entonces una vez inicializado el objeto dos con op1=int y op2 permanece con adv.

Este formato es el que se usa dentro del archivo *dahdi-channels.conf*.

4.7.3.-Gramática de Entidad Compleja

El formato de una entidad compleja es usado por el *iax.conf*, *sip.conf* así como otras entidades, algo importante que debemos comentar es que no todas las entidades manejan el mismo formato, ni tampoco tienen las mismas opciones, cada entidad tienen su propio contexto, pero en este proyecto se tratará la entidad *sip.conf*.

```
[entidad1]
op1=valor1       ;línea 1
op2=valor2       ;línea 2
[entidad2]       ;línea 3
op1=valor3       ;línea 4
op2=valor4       ;línea 5
```

La entidad 1 tienen valores valor1 y valor2 para las opciones op1 y op2. La entidad 2 tiene valores valor3 y valor4 para las opciones op1 y op2.

4.8.-Configuración de Asterisk

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

- Desde el propio CLI (Command. Line Interface)
- Desde los ficheros de configuración (.conf) en /etc/asterisk

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

```
asterisk*CLI>reload
```

4.8.1.-Configuración de sip.conf

En este archivo, se configuran las terminales o extensiones, que utilizaran el protocolo SIP para poder comunicarse y establecer una sesión en el nuestro servidor Asterisk.

En este fichero se definen 3 cosas fundamentales las cuales servirán para identificar las características de las extensiones, como números de extensión, el nombre al que pertenece una extensión, el tipo de extensión, etc.

Partes del fichero sip.conf:

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

Empezaremos con la sección principal o mejor dicho la sección que nunca debe de faltar para que Asterisk funcione, para poder configurar las características de las extensiones, primero debemos saber que existe la sección [general], en esta es donde se definen opciones globales y aspectos por defecto para todas las extensiones SIP.

La sintaxis es la siguiente:

```
[general]
variable1=valor1
variable2=valor2
```

Las opciones más importantes de esta sección son:

VARIABLE	DESCRIPCIÓN
disallow	all : Rechaza todos los codecs (configuración global)
autodomain	Habilita/deshabilita la capacidad de Asterisk de agregar hostnames locales e IP address locales a la lista de dominios, por defecto “no”.
Allow	Permite los codecs en orden de preferencia (use DISALLOW=ALL, antes de permitir otros codecs)
bindaddr	Dirección IP: la dirección IP en la cual escuchar. por defecto 0.0.0.0 (todas los interfaces de red)
bindport	Número: Puerto UDP donde escuchar, por defecto 5060.
callerid	Información de identificación de llamada a utilizar cuando no esté disponible, por defecto “asterisk”.
dtmfmode	Permite especificar el método por el cual se enviarán los tonos ó dígitos pulsados durante la conversación
externip	Dirección IP o un hostname: dirección que vamos a poner en los mensajes SIP si estamos detrás de un NAT. Si un hostname se utiliza como valor, entonces la dirección IP asociada al hostname se resolverá solamente una vez durante la lectura del sip.conf. Si deseas soporte para un hostname asociado a una dirección IP dinámica, utiliza externhost.
externhost	dominio.com
externrefresh	Número: Especifica cuantas veces (en segundos) las operaciones de búsqueda del hostname DNS se deben realizar para el valor configurado en “externhost”. por defecto 10 segundos.
language	Lenguaje por defecto usado por Playback () y Background ().
localnet	Dirección delared/mascaradered: red y máscara locales.
maxexpirey	Número: Duración máxima (en segundos) de las registraciones entrantes que permitimos. Por defecto 3600 segundos.
nat	Informa a Asterisk del tipo de NAT en el que se encuentra
notifyingringing	Notificar la suscripción en estado de RINGING. Por defecto “yes”.
port	Puerto SIP por defecto del peer.
qualify	Comprobar si el cliente esta accesible. Si está en “yes”, los chequeos ocurren cada 2000 milisegundos (2 segundos). Por defecto “no”.
tos	Fijar los parámetros de QoS para el flujo de medios salientes (valores numéricos también se aceptan, como tos=184)
srvlookup	Permitir las operaciones de búsqueda DNS SRV en las llamadas, por defecto “no”.
videosupport	Habilitar el soporte para vídeo SIP, por defecto “no”.
callevents	Cambiar “yes” para recibir eventos en AMI cuando una llamada es puesta en espera.

Tabla 4.9.- Opciones para la sección global “[general]”.

Después de la sección general nos enfocaremos en configurar las características de las extensiones en SIP.CONF correspondientes a Clientes y Servidores. En sip.conf se definen tanto los clientes que se conectarán a Asterisk ya sean estos clientes SIP de software ó softphone, un teléfono SIP físico o un adaptador ATA, así como los proveedores de telefonía SIP que se utilizaran para encaminar llamadas.

Los tipos de extensiones los podemos dividir conceptualmente en:

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

Este archivo utiliza la gramática de entidad compleja, en la cual cada entidad utiliza un contexto como este:

```
[nombre]
Type=xxx
Parametro1=valor
Parametro2=valor
```

Los contextos empiezan siempre con el nombre del contexto dentro de corchetes y termina solamente iniciando otro contexto, debe quedar en claro que todas las variables después de un nombre entre corchetes se asignaran al contexto anterior hasta que se encuentre un nuevo nombre de contexto entre corchetes. Otra forma de definir a un contexto de sip.conf es haciendo referencia al nombre de las extensiones, nombre por el cual se identifican en el DialPlan.

Los parámetros más importantes que deben ser configurados inicialmente son:

PARÁMETRO	DESCRIPCIÓN
type:	peer / friend
context:	Contexto donde entraran las llamadas generadas.
nat:	Indica si el usuario o peer se encuentran tras un nat.
host:	IP remota o dynamic.
username:	Nombre de usuario.
secret:	Contraseña de acceso.
allow y disallow:	Configuraciones de codecs específicas para cada friend/peer.
qualify:	Evalúa el estado del extremo SIP para conocer su accesibilidad y latencia.
register => usuario[:password[:authuser]]@hostservidor[:puerto][/extensión] ESTA OPCIÓN ES ESPECIAL y SIRVE PARA REGISTRAR USUARIOS EN LOS SERVIDORES DE LOS PROVEEDORES DE SERVICIO.	

Tabla 4.10.- Parámetros de sip.conf.

Como ya se explicó la primera fase de esta tesis es la comunicación interna de Luz y Fuerza del Centro sin salir a Internet, y el modelo de red es un modelo centralizado, lo que quiere decir que tendremos un sólo servidor para todas las líneas o extensiones en uso.

Una vez descrito el procedimiento verificaremos nuestro archivo sip.conf.

Empezamos por:

```
[general]
context=default
svrlookup=yes           Búsqueda del servidor DNS
disallow=all            Deshabilitamos todos los codec's
allow=gsm               Primero habilitamos gsm<codec>
allow=alaw              Después habilitamos alaw<codec>
allow=ulaw              Después habilitamos ulaw<codec>
language=es            Cambiamos el lenguaje a español
tos_sip = cs3          Fijamos parámetros para QoS
tos_audio = ef
tos_video = af41
allowguest = yes       Permitimos llamadas de invitados
```

Podemos notar que en esta parte estamos hablando del contexto en el que se encuentra esta etiqueta, hay que recordar que este contexto lo volveremos a observar en extensions.conf.

```
register => nombre.usuario:CONTRASENA@dominio:puerto
```

En la línea anterior se ve como se hace el registro con nuestros proveedores si es que tenemos. En el proyecto no se tiene proveedor de servicios puesto que nos enfocaremos a las comunicaciones internas.

El siguiente paso es dar de alta los usuarios ó extensiones, sería de la siguiente manera:

```
[2000]
type=friend             Nombre del Contexto del usuario 2000
secret=2000            Tipo de usuario
callerid="PRIMER USUARIO" contraseña
qualify=yes           ;Comprobar si el cliente esta accesible cada 2000 milisegundos
nat=no                El cliente no está bajo algún nat.
host=dynamic           La extensión puede conectarse desde cualquier ip.
canreinvite=no
context=default        El contexto al que pertenece en extensions.conf
defaultexpiry=300     tiempo de expiración de registro en segundos.

[2001]
```

```
type=friend
secret=2001
callerid="SEGUNDO USUARIO"
qualify=yes
nat=no
host=dynamic
canreinvite=no
context=default
```

```
[2002]
type=friend
secret=2002
callerid="TERCER USUARIO"
qualify=yes
nat=no
host=dynamic
canreinvite=no
context=default
```

```
[2003]
type=friend
secret=2003
callerid="CUARTO USUARIO"
qualify=yes
nat=no
host=dynamic
canreinvite=no
context=default
```

```
[2004]
type=friend
secret=2004
callerid="QUINTO USUARIO"
qualify=yes
nat=no
host=dynamic
canreinvite=no
context=default
```

```
[ata]
type=friend
host=ip_del_dispositivo_ata
username=ata
secret=CONTRASENA
callerid="nombre_ata"
context=ata
qualify=yes
```

;ejemplo de un adaptador ATA

```
[proveedor]
type=peer
host=xxx.proveedor.com
username=nombre.usuario
fromuser=nombre del usuario
secret=CONTRASENA
context=proveedor
insecure=port,invite
```

;ejemplo de algún proveedor

;la URL de nuestro proveedor

```
qualify=yes  
canreinvite=no
```

Hasta este momento tenemos configurado nuestro archivo sip.conf de una manera básica y sin que todavía podamos establecer ninguna llamada, ya que nos faltaría complementar a nuestras extensiones de una metodología a seguir al momento de ser invocadas por cualquier otra extensión, para ello necesitamos crear reglas de marcado para comunicarle al núcleo de Asterisk como debe tratar dichas extensiones y así el núcleo sabrá con exactitud cómo responder a cada una de las extensiones, pudiera ser que alguna extensión este inhabilitada, y es donde el Dial Plan o Plan de Mercado entraría en acción para saber cómo debe corresponder a dicha solicitud.

4.8.2.- Dial plan o Plan de Mercado

Antes de comenzar debemos dejar en claro ciertos conceptos para así poder entender mejor al Plan de Mercado.

Como su nombre lo indica se trata de la configuración de la centralita Asterisk que indica el itinerario que sigue una llamada desde que entra o sale del sistema, hasta que llega a su punto final. Se trata del comportamiento lógico de la centralita.

Con este concepto podríamos aventurarnos a decir que es corazón de Asterisk (*). El Dial Plan está definido en el fichero extensions.conf. Y su arquitectura puede expresarse en el siguiente esquema:

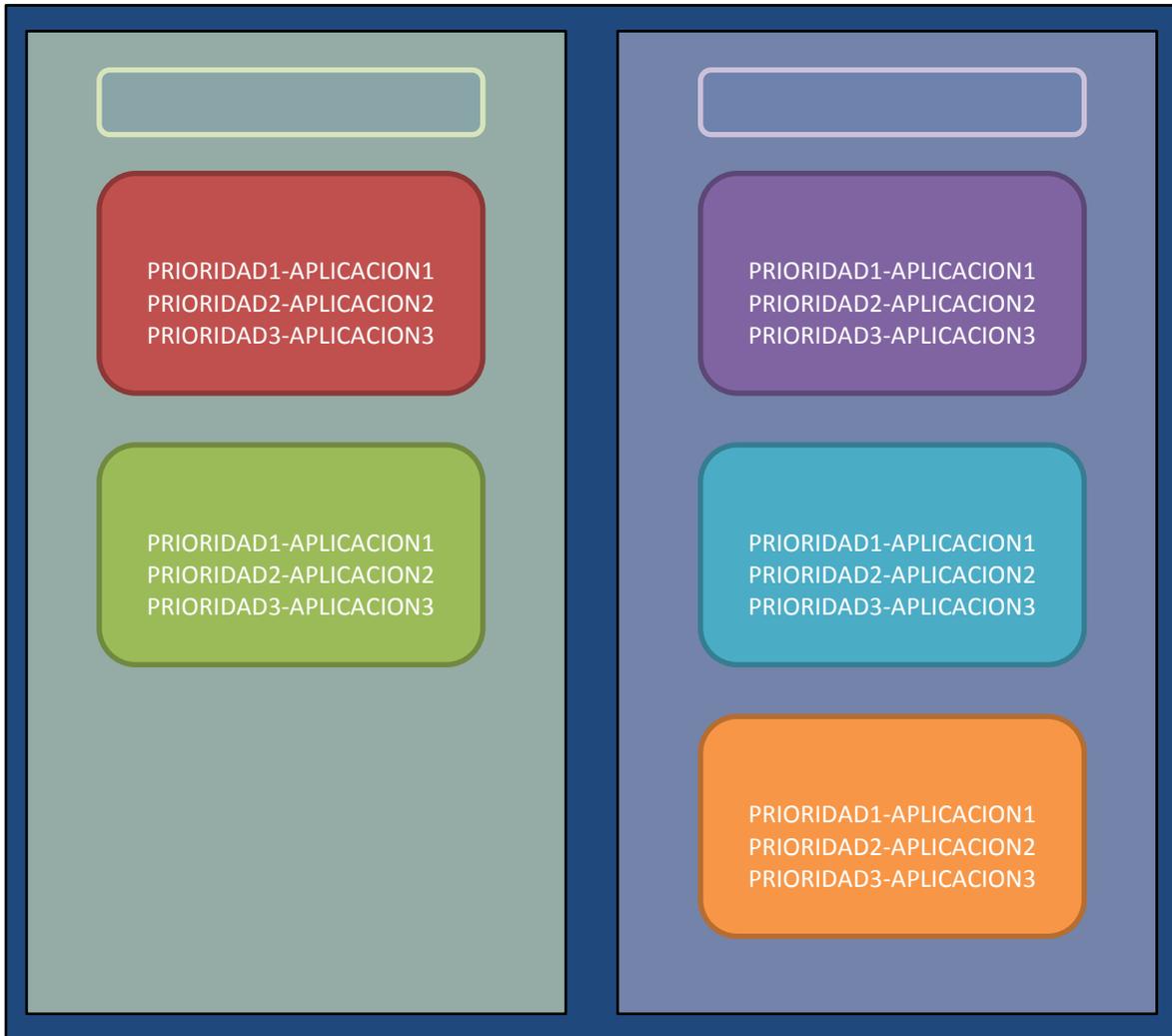


Figura 4.21.-Arquitectura del Dial Plan.

4.8.2.1.-Extensiones

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

Las extensiones se acceden cuando:

- Se recibe una llamada entrante por un canal dado (SIP, IAX2, H323, etc.).
- El usuario que ha llamado marca la extensión.
- Se ejecuta un salto de extensiones desde el Dial plan de Asterisk.

Ya hablamos de contexto en sip.conf, pero en el Dial Plan es más bien la lógica de comportamiento, una colección de extensiones las cuales siguen un patrón. El Dial Plan se divide en uno o varios contextos:

Y funcionan de varias maneras como por ejemplo:

- Para poder diferenciar el 'lugar' donde se encuentra una llamada.
- Aplicar políticas de seguridad.
- Menús y submenús diferenciados.
- Etc.

4.8.2.2.-Aplicaciones

Otro concepto importantísimo para utilizar el Dial Plan es a lo que llamamos Aplicación que en inglés es “Application”. En el Dial Plan, Asterisk realiza la ejecución secuencialmente de los comandos asociados a cada una de las extensiones. A estos comandos se les conoce como aplicaciones, dichas aplicaciones controlan el comportamiento de cualquier llamada y del sistema en sí.

Las *aplicaciones generales* más comunes son:

NOMBRE	DESCRIPCIÓN
Wait (n)	Espera n segundos, ignorando los dígitos marcados durante.
WaitExten	Espera n segundos, pero gestionando los dígitos marcados.
WaitMusicOnHold	Reproduce música en espera durante n segundos.

Tabla 4.11.- Aplicaciones Generales.

Las *aplicaciones de gestión de llamadas* más comunes son:

NOMBRE	DESCRIPCIÓN
Answer	Sintaxis: Answer([delay]) Contesta un canal si este está timbrando. Puede recibir opcionalmente el parámetro delay que le indica si debe esperar un número determinado de milisegundos antes de contestar.
Hangup	Sintaxis: Hangup(). Cuelga el canal y retorna -1
Ringin	Envía la señal de tono de llamada.
Dial	Sintaxis: Dial(type/identifier, timeout, options, URL) Este comando es el más importante. Marca un canal especificado y lo vincula con el canal corriente.
Busy	Envía la señal de ocupado al origen.

Tabla 4.12.- Aplicaciones de Gestión de Llamadas.

Las aplicaciones de control de flujo más comunes son:

NOMBRE	DESCRIPCIÓN
Goto	Sintaxis: Goto([[context]extensión]priority) Salta la ejecución del plan de marcado a un contexto, extensión y prioridades dados. Si solo se pasa un parámetro se sobreentiende que se trata de una prioridad dentro del mismo contexto.
GotoIf	Sintaxis: Gotoif(condición ? prioridad 1 : prioridad 2) Salta la ejecución del plan de marcado a un contexto, extensión y prioridades dados. Salta a la prioridad 1 si la condición se cumple. Salta a la prioridad 2 si la condición no se cumple.
GotoIfTime	Sintaxis: GotoIfTime(<times> <weekdays> <mdays> <months>?[[context]exten]priority)

Tabla 4.13.- Aplicaciones de Control de Flujo.

Las aplicaciones de reproducción de sonidos más comunes son:

NOMBRE	DESCRIPCIÓN
Background	Sintaxis: Background(filename1[&filename2...][opciones[langoverri]context]) Reproduce un listado de archivos de audio, cuando termina de reproducir el último archivo termina su ejecución por lo que si se quiere seguir esperando por una extensión se debe usar otra aplicación en conjunto llamada WaitExten. Esta aplicación es típicamente usada para reproducir el mensaje de bienvenida de un IVR. La aplicación Background puede recibir algunas opciones: s: Saltarse la reproducción del audio si el canal no se encuentra contestado n No contestar el canal antes de reproducir los archivos m Dejar de reproducir el audio si un dígito marcado coincide con una extensión.
Playback	Sintaxis: Playback(filename1[&filename2...][,options]) Reproduce uno o más archivos de audio. La diferencia con la aplicación Background es que Playback reproduce todo el archivo de audio hasta el final y no retorna el control hasta que termina la reproducción. Nota: No es necesario especificar la extensión del archivo de audio en esta función. Los archivos de audio son buscados en el directorio /var/lib/asterisk/sounds.
SayDigits	SayDigits (dígitos) Reproduce los dígitos

Tabla 4.14.- Aplicaciones de Reproducción de Sonido.

Para obtener una lista completa de aplicaciones disponibles deberás de utilizar el siguiente comando en la línea de comandos de Asterisk.

```
CLI>show applications
```

ó

```
CLI>show application <nombre>
```

Estos y más comandos los veremos más adelante.

4.8.2.3.-Contexto

Un contexto es la forma de decir que las extensiones declaradas debajo de los corchetes [xx] son pertenecientes al contexto xx por lo que podemos definir contexto como la agrupación de instrucciones de una extensión declarada en el DialPlan.

La sintaxis de un contexto en el DialPlan sería:

```
[contextoxxx]
exten => 100,1,PlayBack(canción)
exten => 100,2,Dial(SIP/nombre extensión)
```

Hay que recordar que para que esta extensión pueda funcionar debe ser declarada en sip.conf y se le debe especificar que pertenece al contexto=contextoxxx así:

```
[usuario1]
type=friend
....
context=contextoxxx
```

4.8.2.4.-Prioridad

La prioridad es una parte fundamental en el Dial Plan ya que está indica la secuencia en que los pasos se irán ejecutando, esta prioridad empieza regularmente en 1. La prioridad “n”, es equivalente a sumar uno a la prioridad anterior. Esto permite editar y modificar rápidamente el Dial Plan.

La sintaxis sería:

```
[contextoxxx]
exten => 100,1,PlayBack(canción)
exten => 100,n,Dial(SIP/nombre extensión)
```

Siendo la línea seguida de la extensión, el primer paso a seguir para el Dial plan, los números siguientes deben de ser mayores, en este caso es una “n” la cual tiene como valor un “2”.

4.8.3.-Dial

Como lo mencionamos en el listado de aplicaciones, Dial es la aplicación más importante, es la que se encarga de enlazar las extensiones, por lo que tendremos que darle mucha importancia a la misma y analizarla a detalle.

La aplicación Dial sirve para realizar una llamada a un determinado destino, si el destino acepta la llamada, Asterisk se encarga de unir el origen primario de la llamada con destino de la misma.

Algo muy importante de esta aplicación es que no importa la tecnología o protocolos utilizados. Un usuario SIP puede llamar a otro usuario IAX2.



Figura 4.22.- Flujo de la Llamada con Aplicación Dial.

La sintaxis depende del canal, para este proyecto se utilizara canales SIP y la sintaxis del comando Dial para el canal SIP es:

Sintaxis:

Dial (Tecnología / identificador | timeout | opciones | URL)

- Tecnología: SIP, IAX2, ZAP, MISDN,...
- Identificador: Es el destino (número o nombre) que se llamará por el canal.
- Timeout: Tiempo máximo de espera.
- Opciones.
- URL: Envía una URL al destino, si lo soporta.

Las opciones más comunes que Dial acepta son:

- t: permite que la llamada sea transferida por el destino.
- T: permite que la llamada sea transferida por el origen.
- m: reproduce música en espera mientras el destino acepta la llamada.

Los identificadores son la forma en que vamos a decirle al Asterisk que se trata de un canal tipo SIP con número de extensión x y la sintaxis es la siguiente:

```
Dial(SIP/202)
Dial(SIP/201|100|t)
Dial(SIP/944416024@proveedor)
```

Para hacer coincidir muchas extensiones en una sola instrucción debe de haber Patrones de Coincidencia estos patrones depende de la planeación de extensiones, muchas extensiones comienzan por cierto número y al querer modificar cada una en el fichero extensions.conf nos costaría mucho trabajo, así que para eso tenemos reglas, las cuales evitan tener que editar una a una cada extensión, haciendo de una sola instrucción la madre de todas las extensiones tenga algún patrón de coincidencia.

Para indicar patrones, se utiliza el carácter “_” y la sintaxis pueden utilizarse de la siguiente manera:

- X: Indica un dígito del 0 al 9
- Z: Indica un dígito del 1 al 9
- N: Indica un dígito del 2 al 9
- [129] Indica el 1, 2 o 9
- “.” Indica uno o más caracteres.

Un ejemplo podría ser:

```
exten=> _7XXX
exten=> _00.
```

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

La sintaxis de una variable es:

```
${variable}
```

Los tipos de variables son:

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

Para asignar la variable se sigue el siguiente formato:

```
SetVar(Variable=valor)
```

```
SetGlobalVar(Variable=valor)
```

Las cadenas son esenciales en cualquier lenguaje, para asterisk también y se pueden aplicar de la siguiente manera:

```
Subcadenas: ${Variable : offset : longitud } ; Devuelve la subcadena de variable que comienza en offset y con la longitud especificada.
```

```
${ 123456789:2:3 } ;devuelve 345
```

```
${Variable1}${Variable2} ;Concatenación de variables
```

También existen las variables de canal definidas automáticamente por cada canal, las más comunes son:

```
${CALLERID}: Caller ID actual, nombre y número.
```

```
${CONTEXT}: Contexto actual.
```

```
${EXTEN}: Extensión actual.
```

```
${CHANNEL}: Canal actual.
```

```
${DIALSTATUS}: Estado de la llamada: unavailable, congestión, busy, noanswer, answer, cancel, hangup.
```

```
${DATETIME}: Hora actual.
```

Las expresiones para realizar alguna operación lógica, de comparación o aritmética tienen su Sintaxis y es:

```
$(expr1 operador expr2)
```

- Operadores Lógicos: |(or) , &(AND)
- Operadores de Comparación: =, !=, <, >, <=, >=
- Operadores Aritméticos: +, -, *, /, %

```
exten => 1,1,SetVar(total=${1 + 1}) ;operador aritmético  
exten => 1,2,GotoIf($${CALLERID}=123456)?10:20) ;operador de comparación
```

En este esquema se puede apreciar que para cada contexto existen una o más extensiones, las cuales funcionan de diferente manera según sus aplicaciones que utilicen.

4.8.4.- Configuración de extensions.conf

Una vez entendidos los conceptos anteriores y su sintaxis, empezaremos con la sección general del DialPlan, donde pondremos las opciones que valen para todo el fichero.

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

La sección general es la sección que no debe faltar nunca en una buena configuración de Asterisk, con static=yes y writeprotect=no, se habilita salvar un plan de marcado a través del comando:

```
Asterisk*CLI> 'save dialplan'.
```

La opción autofallthrough=yes, hace que al finalizar las tareas de Asterisk este pueda terminar la llamada con BUSY, CONGESTION o HANGUP.

Clearglobalvars realiza la persistencia de las variables globales para que no sean eliminadas con un reload en el CLI.

Enseguida de la sección general pondremos la sección de globals:

```
[globals]
```

En globals se pueden definir las variables globales que pueden usarse posteriormente en las extensiones.

Una vez declaradas las secciones generales y globales tendremos que comenzar a darle forma a la verdadera configuración de Asterisk, y comenzaremos creando el contexto default.

```
[default] ;Contexto con el nombre de default
```

Una vez dentro del contexto lo primero que será añadir una extensión de prueba y que reproduzca un mensaje, esto quiere decir que no es necesariamente una extensión que hayamos dado de alta en el archivo sip.conf, más bien es una extensión para realizar una prueba con la aplicación “echo”, la cual sirve para escuchar lo que hablamos y saber que Asterisk está en funcionamiento. Y la segunda extensión va a realizar una espera de 5 segundos antes de colgar, esta es la manera de probar si Asterisk está en funcionamiento.

```
exten => 600,1,Answer() ;la extensión es contestada
exten => 600,2,Playback(demo-echotest) ;reproducir un sonido
exten => 600,3,Echo() ;prueba de echo
exten => 600,4,Playback(demo-echodone)
exten => 600,5,Hangup() ;colgar

exten => 601,1,answer()
exten => 601,2,wait(5) ;espera de 5 segundos
exten => 601,3,hangup() ;colgar
```

Las extensiones que dimos de alta en sip.conf deben de coincidir con la configuración del siguiente texto.

```
; Extensión 2000 ;Comentario
exten => 2000,1,dial(SIP/2000,30,tT) ;Prioridad 1
exten => 2000,2,Hangup() ;Prioridad 2

; Extensión 2001
exten => 2001,1,dial(SIP/2001,30,tT) ;Extensión de tipo SIP/numero 2001
exten => 2001,2,Hangup() ;colgar
```

```

; Extensión 2002                                     ; 30 seg. de espera si no es contestada
exten => 2002,1,dial(SIP/2002,30,tT)                   ;Prioridad n=2
exten => 2002,n,Hangup()

; Extensión 2003                                     ;opciones tT para transferir llamadas
exten => 2002,1,dial(SIP/2002,30,tT)
exten => 2002,2,Hangup()

; Todas en una siguiendo un patrón
exten => _2XXX,1,Dial(SIP/${EXTEN},50,tT)
exten => _2XXX,2,Hangup()

```

4.9.-Administración básica del CLI de Asterisk.

Asterisk command line *interface* (CLI) es la interfaz en modo consola por la cual nos podremos comunicar con Asterisk, y saber todo lo que está pasando con nuestra central telefónica.

Podemos conectarnos con:

```
Asterisk # asterisk -r
```

Una vez dentro de la interface de comando nos aparecerá en siguiente promp:

```
nombre_de_su_sistema*CLI>
```

Dentro del CLI se puede gestionar el sistema ayudándonos de una serie de comandos, podemos verificar la versión con la que estamos trabajando:

```
nombre_de_su_sistema*CLI> show version
```

Al modificar el archivo extensión (Dial Plan) debemos cargar la nueva configuración del sistema, esto lo haremos mediante el comando reload.

En primer lugar cargaremos la configuración del fichero extensions.conf:

```
nombre_de_su_sistema*CLI> extensions reload
```

ó

```
nombre_de_su_sistema*CLI> dialplan reload
```

Es posible analizar cómo ha interpretado Asterisk el fichero extensions.conf, con esto confirmamos posibles errores de sintaxis u otros posibles.

```
nombre_de_su_sistema*CLI> show dialplan <contexto>
```

Para cargar la configuración del archivo sip.conf solo debemos de recargarlos con el comando:

```
nombre_de_su_sistema*CLI> sip reload
```

4.10.-Verificación de servicios vía consola.

Para verificar los peers y friends asociados al sistema Asterisk debemos ejecutar los siguientes comandos

```
nombre_de_su_sistema*CLI> sip show users ;Para ver los usuarios del sip
```

```
asterisk*CLI> sip show users
Username          Secret          Accountcode     Def.Context     AC
L Nat
2003              RFC3581         2003            default         No
2002              RFC3581         2002            default         No
2001              RFC3581         2001            default         No
2000              RFC3581         2000            default         No
asterisk*CLI> _
```

Figura 4.23.- Verificación vía consola.

```
nombre_de_su_sistema*CLI> sip show peers ;Para ver los “peers” definidos
```

De este modo podremos confirmar que todos los peers que hemos declarado anteriormente se han registrado con el sistema Asterisk de forma satisfactoria

```
asterisk*CLI> sip show peers
Name/username     Host            Dyn Nat ACL Port    Status
2003              (Unspecified)  D      0      UNKNOWN
2002              (Unspecified)  D      0      UNKNOWN
2001/2001         192.168.1.90   D      31010 OK (101 ms)
2000/2000         192.168.1.88   D      62394  OK (105 ms)
4 sip peers [Monitored: 2 online, 2 offline Unmonitored: 0 online, 0 offline]
asterisk*CLI> _
```

Figura 4.24.- Verificación de usuarios online.

CAPITULO 5

INSTALACIÓN Y CONFIGURACIÓN DE SOFTPHONE PARA ASTERISK (CLIENTE)

5.1.- Softphone

El siguiente paso será instalar los softphone de los clientes pero para eso debemos dejar en claro que son y porque decidimos usar estos como la solución mas económica para este proyecto de implementación.

5.1.1.- ¿Qué son?

En el mundo de Voz sobre IP existe una lista interminable de hardware como tarjetas, teléfonos IP, ata, etc. pero también existen software que facilitan la función de de algunos de ellos. Existe uno en específico, y se llama Softphone que es el software que realiza una simulación de teléfono en una computadora, permitiendo así la comunicación con otras computadoras que posean este mismo tipo de software.

Los softphone tienen tres maneras de clasificarse opensource, pagos y gratuitos. Por lo general, existen versiones comerciales y gratuitas de la mayoría de softphones, pero estas últimas generalmente carecen de características importantes, entre ellas los codecs comerciales g729 y g723.

Algunos ejemplos de estos son:

WengoPhone es una alternativa de código abierto a aplicaciones como Skype, que es software propietario, aunque su funcionamiento es similar. Openwengo, disponible para Windows, Linux y Mac OS X, es el código fuente y está disponible bajo licencia GPL y que nos permite establecer llamadas VoIP empleado el estándar abierto SIP.



Figura 5.1.- Softphone WengoPhone

Gizmoes otro de las muchas opciones, Gizmo utiliza SIP como estándar para las comunicaciones de voz. De esta manera puedes comunicarte con cualquier red o programa de telefonía IP que soporte este estándar.



Figura 5.2.- Softphone Gizmoes

SightSpeed: es un Softphone para la comunicación la cual implementa la videoconferencia la cual te permite establecer sesiones de video comunicación.



Figura 5.3.- Softphone SighSpeed

X-Lite 3.0 es la solución gratuita para establecer una llamada IP en redes basadas en Asterisk y está basado en la arquitectura del eyeBeam 1.5 (cliente telefónico pago de la misma empresa). Tiene ciertas capacidades, como calidad de audio y video superior, y una excelente libreta de direcciones.



Figura 5.4.- Softphone Xlite.

SJPHone fue exitosamente utilizado en cientos de servicios de telefonía IP por todo el mundo, desde pequeñas pymes haces grandes telecoms. Cabe destacar la posibilidad de adaptar la interfaz a cualquier empresa, traducida a varios idiomas configuración y soporte remoto, soporte para MAC OS, Linux, Windows Móvil.



Figura 5.5.- Softphone SJPHone.

Zoiper(antes IDEFISK), es un softphone SIP e IAX compatible con la plataforma Asterisk y otros sistemas PBX. La edición gratuita de ZOIPER incluye características básicas para protocolos SIP e IAX/IAX2, aunque la versión comercial aumenta su soporte.



Figura 5.6.- Softphone Zoiper.

5.1.2.- ¿Por qué Softphone para el proyecto?

Se eligió el uso de software en lugar de el uso de teléfonos IP por el elevado costo de inversión inicial, lo que se busca es crear un ambiente propicio para la implementación de voz sobre IP, para lo cual se incluirá exclusivamente software en la primera etapa de la implementación, para posteriormente realizar una segunda etapa y montar los teléfonos IP físicos en cada una de las extensiones que en este momento están funcionando.

El softphone nos provee de varias ventajas las cuales son:

- Fácilmente configurable para los cambios de oficina.
- Para hacer almacenaje de datos.
- Para hacer manejo de e-MAIL.
- Para hacer y/o recibir llamadas VoIP en cualquier momento.
- Fácil de hacer videoconferencias de persona a persona.

5.2.- Instalación de Xlite 3.x

Escogimos Xlite por su fácil manejo y su amplia gama de configuraciones, hace que sea la solución más adecuada para la implementación de teléfonos por software en Luz y Fuerza del Centro, empezaremos con su instalación en un ambiente Windows ya que es el software que se maneja en la empresa.

Primero descargaremos la versión más reciente, la podemos encontrar en la página oficial <http://www.sipquest.com/>.

Una vez descargado empezaremos la instalación.

Nos aparecerán estas pantallas, la configuración por default está bien así que con que demos clic en siguiente está bien.

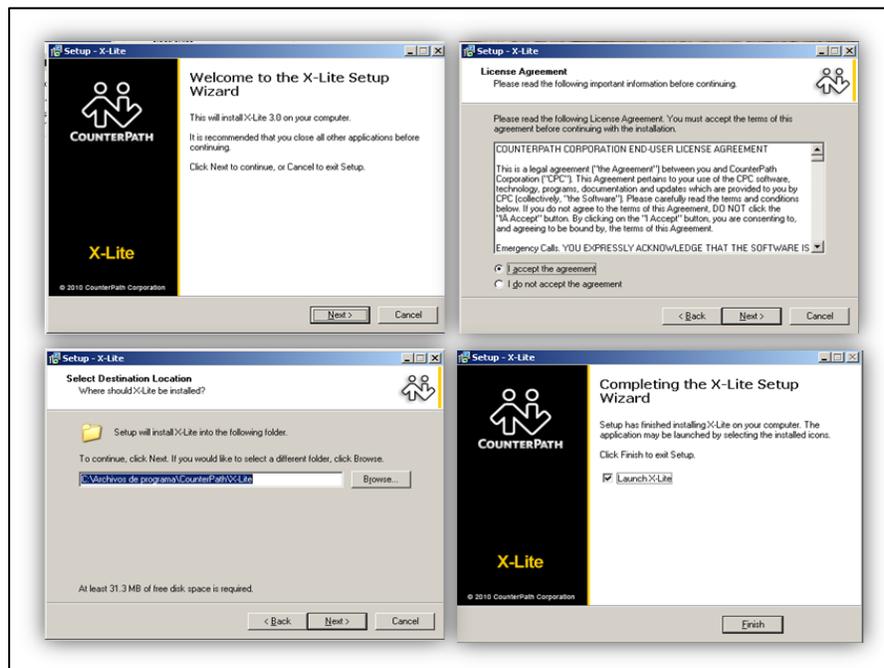


Figura 5.7.- Instalación de Xlite.

5.3.- Configuración de Xlite

La configuración de Xlite es de manera fácil y solamente necesitamos tener la siguiente información:

- # de extensión: es el número que pusimos en el archivo sip.conf
- Usuario: Es la persona a la que pertenece el numero de extensión.
- Servidor: Es el numero de IP del servidor Asterisk.

Una vez dentro del programa Xlite iremos a la sección de SIP Accounts y ahí es donde configuraremos nuestra cuenta que tenemos en el servidor Asterisk. Haciendo clic en add.

Una vez terminada la configuración pondremos aceptar y listo ya tenemos nuestra cuenta creada del lado del cliente. Esto hará que Xlite se registre en el servidor Asterisk haciendo que en la pantalla de Xlite aparezca la siguiente leyenda:



Figura 5.8.- Pantalla de Xlite al estar registrado correctamente.

5.4.- Verificación de intercomunicación entre clientes.

Para hacer la verificación de los servicios tendremos que hacer una llamada de prueba entre extensiones existentes y deberá aparecer en la pantalla de la extensión llamada un mensaje de que se está realizando una llamada de la extensión llamante. Como se muestra en la figura 5.8.



Figura 5.9.- Pantalla de Xlite al llamar entre extensiones.

Una vez que los softphones estén instalados se procederá a corroborar su funcionamiento con un analizador del tráfico de red, este se hará desde alguna máquina cliente.

Debido a que en los trabajos realizados en LYF es posible que en algún momento se tenga que trabajar bajo mucha carga de red, voy a mostrar las pruebas de los dos posibles casos de utilización de ancho de banda, la primera cuando está todo el ancho de banda disponible y la segunda cuando se esté trabajando bajo mucha demanda de red de parte del cliente. Se hicieron un gran número de pruebas pero para cuestión de representación solo mostrare una de cada caso.

Estas llamadas fueron monitoreadas por medio de Wireshark en el momento en que se realizaron las pruebas.

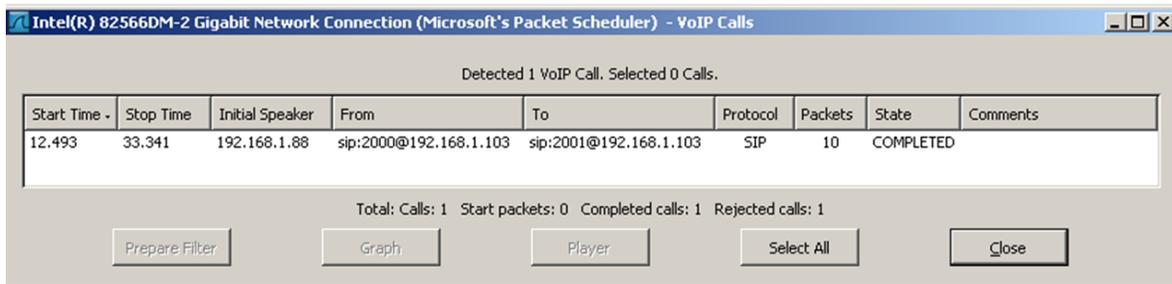


Figura 5.10.- Llamada de voz IP (caso 1).

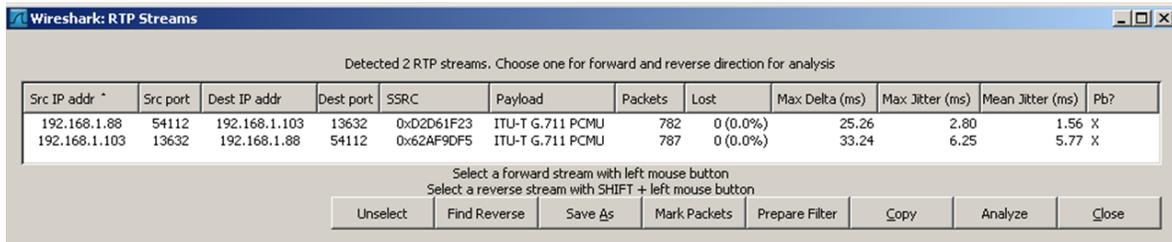


Figura 5.11.- Características de la llamada de voz IP (Caso 1).

En la figura anterior se muestran las características de la llamada realizada sin ningún tráfico en la red. En ella se muestran las direcciones IP de origen y destino de la llamada. Así como las siguientes características:

Característica	Descripción
Tipo de códec utilizado en la conversación	ITU-T G.711 PCMU
Total de paquetes	782 y 787
Paquetes Perdidos	0 y 0
Retraso Máximo	25.26 ms y 33.24 ms

Jitter Máximo	2.80 ms y 6.25 ms
Jitter promedio	1.56 ms y 5.77 ms

Tabla 5.1.- Características de la llamada IP

Analizando el desglose de paquetes enviados en la figura 5.11 abajo mostrada se puede ver los paquetes de origen a destino, también se pueden analizar los paquetes en sentido opuesto del destino al origen de la llamada IP.

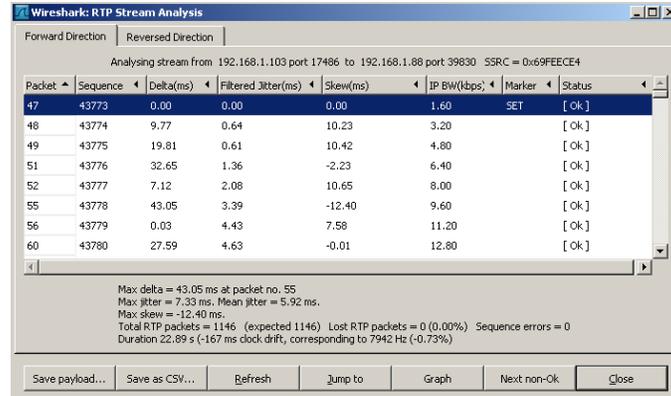


Figura 5.12.- Desglose de paquetes enviados (caso 1).

En la figura 5.12 se muestra el flujo de la sesión en la llamada establecida, como se dijo en el capítulo 3 este establecimiento se hace como peer to peer entre dos UA (agentes de usuario), en esta grafica se muestra el resultado que arroja Wireshark sobre las llamadas VoIP.

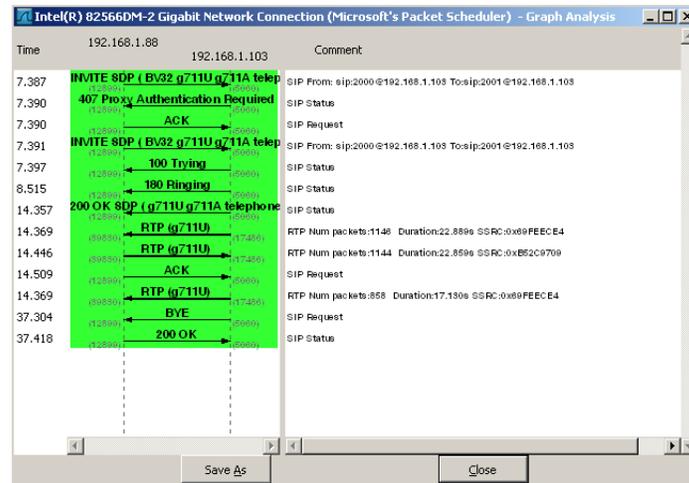


Figura 5.13.- Flujo de establecimiento de una sesión (caso 1).

La siguiente figura es consecuencia del caso 2 donde se realizó la llamada desde una computadora cliente, la cual con propósito de prueba se le generó ocupación de ancho de banda, esto quiere decir que en el momento de la llamada, esta computadora mantuvo una constante carga de procesos de red, como descargas vía Internet, descargas vía Ethernet. Estas acciones con el fin de consumir ancho de banda para poner a prueba nuestra llamada y hacer una simulación del trabajo habitual en una oficina dentro de Luz y Fuerza del Centro.

Figura 5.14.- Características de la llamada de voz IP (Caso 2).

Src IP addr	Src port	Dest IP addr	Dest port	SSRC	Payload	Packets	Lost	Max Delta (ms)	Max Jitter (ms)	Mean Jitter (ms)	Pb?
192.168.1.88	6296	192.168.1.103	10222	0x1D84134F	ITU-T G.711 PCMU	2990	0 (0.0%)	29.35	1.37	0.49	X
192.168.1.103	10222	192.168.1.88	6296	0x53399EBF	ITU-T G.711 PCMU	2980	0 (0.0%)	39.46	7.44	6.03	X

En la figura 5.13 se muestra los resultados de la llamada en el caso 2 donde se sometió a una demanda de ancho de banda para así poder comparar los valores que nos arroja Wireshark, dando como resultado algo muy similar al caso uno.

Y para terminar la última prueba de verificación sería la opción de telefonía-SIP la cual nos muestra todos los paquetes que fueron analizados en la máquina cliente así como las respuestas que obtuvo del servidor Asterisk.

Category	Sub-category	Count
Informational SIP 1xx	SIP 180 Ringing	1
	SIP 100 Trying	3
	SIP 200 OK	28
Client errors SIP 4xx	SIP 401 Unauthorized	10
	SIP 404 Not Found	9
	ACK	1
INVITE	1	
SUBSCRIBE	18	
OPTIONS	25	
REGISTER	2	
BYE	1	

Average setup time 9843 ms
Min 9843 ms
Max 9843 ms

Figura 5.15.- Estadística SIP.

CONCLUSIONES

El principal objetivo de esta tesis fue la implementación de VoIP (Asterisk) para la comunicación interna de Luz y Fuerza del Centro. A través de lo investigado y de la información adquirida, podemos dar pie a la factibilidad de la inserción de esta tecnología a la empresa puesto que como se planteo es una opción de menor costo a la actual forma de intercomunicación de la entidad.

Dentro de la investigación se pudo demostrar comparando costos de inversión las bondades de esta tecnología, se muestra que muchas empresas están migrando sus sistemas actuales a sistemas con tecnología IP, dado que sus prestaciones son muchísimas y a comparación de la actual forma de comunicación es mucho más barata.

También se observó que se necesita de capacitación práctica para el uso de nuevas tecnologías en la entidad, esto debido a la falta de preparación tecnológicamente hablando de la entidad.

Otros aspectos importantes identificados fueron los pros y los contras de implantar esta tecnología, ya que si bien esta tecnología tiene más pros que contras, se debe estudiar bien las desventajas de dicha tecnología para el caso específico de cualquier otra empresa.

En base a la información obtenida, se determino que los recursos necesarios para iniciar el proyecto son prácticamente nulos, ya que se reutilizara la infraestructura de la empresa como su red de datos, donde está incluida una de las más grandes redes de fibra óptica de nuestro país. En cuestión de equipos de comunicación, como se planteo anteriormente de ocuparan teléfonos por software ó mejor conocidos como Softphones, los cuales están disponibles sin ningún costo en la red (no todos).

Otro de los factores por lo que se plantea que Voip es una tecnología adecuada para una empresa de la talla de Luz y Fuerza del Centro es que cada vez mas está creciendo la cantidad de datos que se pueden transmitir vía internet o mejor dicho vía protocolo IP por tal motivo se cree que esta tecnología a corto y mediano plazo desplazara a la actual forma de comunicación en todo el país, y por ende en todas las demás empresas, dando pie a una revolución en las comunicaciones. Y si empezamos en Luz y Fuerza del Centro a emplear esta tecnología desde este momento llegara el punto en que la empresa no sufrirá de una etapa de transición, y al llegar el futuro tengamos bases y posibilidades de competencia con cualquier empresa.

También se determinó la factibilidad de que esta tecnología sea la única disponible en la empresa dejando atrás a la actual, que aparte de ocupar otra infraestructura, el mantenimiento de la misma es por medio externo, esto quiere decir que se necesitan de terceros para mantener esta red de telefonía funcionando y en mantenimiento dando pie a más inversión y menos resultados, en cambio si la tecnología que en esta tesis se presenta llegase a ser la única forma de comunicación en la entidad, la inversión de su mantenimiento será la misma inversión que actualmente se hace a la red de datos puesto que se plantea que sea la misma red de datos la que se utilizaría para transmitir la voz.

En las proyecciones realizadas en el ámbito financiero se observó que la tecnología VoIP es una inversión rentable a corto plazo, debido a la naturaleza de su operación ya que es crucial para el buen desempeño de las labores dentro de la empresa. Las cuales reflejan los beneficios financieros a partir prácticamente después de su implementación.

Debido a lo investigado se puede concluir que VoIP es la mejor opción para la transmisión de voz de manera interna no solo en Luz y Fuerza del Centro, sino también en cualquier empresa que posea una red de datos. Se mostró que Voip es capaz de brindar un servicio de igual o mejor manera que las empresas especializadas.

BIBLIOGRAFÍA

Libros

- 1.- Denise Donohue, "Voice Gateways and Gatekeeper", Cap 9 Voice Over IP, Prentice Hall, 1999.
- 2.- CAVE, Hellis. Conferencia Magistral Simposium Internacional de Telematica Inter Voice Brite Inc. 2001.
- 3.- Osland P. y Dinh, K. Perceived Voip quality under varying traffic conditions, 17th Nordic Teletraffic Seminar, Norway, 2004.
- 4.- Flavio Goncalves, Asterisk™ PBX, Guía de Configuración, Rio de Janeiro, Edit. Independiente, 2007
- 5.- Jim Van Meggelen, Jared Smith and Leif Madsen, Asterisk The Future of Telephony, Beijing, O'Reilly Media, Inc.,2005.
- 6.- Alberto EscuderoPascual, Louise Berthilson,VoIP para el desarrollo,Edit independiente,2007.
- 7.- Daniel Collins, Carrier Grade Voice Over IP, McGraw-Hill Professional Publishing, USA,2000.
- 8.- Alan B. Johnston, SIP: Understanding the Session Initiation Protocol, Artech House, 2001.
- 9.- Vineet Kumar, Markku Korpi y Senthil Sengodan, IP Telephony with H.323: Architectures for Unified Networks and Integrated 10.- Services, John Wiley & Sons, 2001.
- 11.- Jonathan Davidson, James Peters, Manoj Bhatia, Satish Kalidindi, Sudipto Mukherjee, Voice over IP Fundamentals (2nd Edition),USA, 2000.

Sitios web

- 1.- Conectividad VoIP, http://www.osmosislatina.com/conectividad/tipos_de_conexion.htm#cswitched [Consulta: Diciembre 2009]
- 2.- Historia de VoIP, <http://voipex.blogspot.com/2006/04/historia-de-voip.html>, [Consulta: Diciembre 2009]
- 3.- Historia de Asterisk, <http://bytecoders.homelinux.com/content/historia-de-asterisk-pbx.html>, [Consulta: Enero 2010]
- 4.- Lógica básica de la voz sobre IP, <http://www.scribd.com/doc/14764672/VOIP>, [Consulta: Enero 2010].
- 5.- Precios Telmex México, <http://www.telmex.com/mx/negocio/telefonía/local-red-multilinea.html>, [Consulta : Agosto 2010].
- 6.- Factibilidad de uso, http://www.fonaes.gob.mx/pdf/capacitacion/estudio_de_factibilidad.pdf,[Consulta: Enero 2010].
- 7.- Licencias de Software, <http://www.agendadigital.cl/files/Uso%20de%20Software%20Libre%20en%20el%20Estado.pdf>, [Consulta : Enero 2010]
- 8.- Telefonía IP, <http://www.lsidotcom.com/productos/telefonía/intro>, [Consulta : Enero 2010]
- 9.- Comunicaciones, <http://es.wikipedia.org/wiki/Ethernet>, [Consulta: Febrero 2010]
- 10.- Skype, <http://www.oobo.com/origen-nombre-skype.html>, [Consulta : Febrero 2010]
- 11.- Conmutación de Circuitos, http://www.microsoft.com/windows/windows2000/es/advanced/help/sag_TAPIconcepts_116.htm, [Consulta : Febrero 2010]
- 12.- Codec's, <http://www.voipforo.com/codec/codecs.php>, [Consulta: Febrero 2010].
- 13.- Protocolos, http://es.wikipedia.org/wiki/Protocolo_%28inform%C3%A1tica%29, [Consulta : Febrero 2010]
- 14.- SIP, <http://objetoide.blogspot.com/2008/06/protocolo-sip-qu-es-y-para-qu-sirve.html>, [Consulta : Febrero 2010].