

**UNIVERSIDAD NACIONAL DE INGENIERÍA**

**FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA**



**IMPLEMENTACION DE UNA CENTRAL PRIVADA POR SOFTWARE  
USANDO UN SERVIDOR ASTERISK CON ACCESO A LA RED  
PÚBLICA DE TELEFONIA CONMUTADA Y ABONADOS SIP**

**INFORME DE SUFICIENCIA**

**PARA OPTAR EL TÍTULO PROFESIONAL DE:**

**INGENIERO ELECTRÓNICO**

**PRESENTADO POR:**

**CARLOS ENRIQUE ROJAS SOSA**

**PROMOCIÓN  
2004 - II**

**LIMA – PERÚ  
2009**

**IMPLEMENTACIÓN DE UNA CENTRAL PRIVADA POR SOFTWARE USANDO  
UN SERVIDOR ASTERISK CON ACCESO A LA RED PÚBLICA DE TELEFONÍA  
CONMUTADA Y ABONADOS SIP**

## **DEDICATORIA**

... El presente trabajo está dedicado a mis padres Benjamín y Teresa, a mi familia y a mis amigos, quienes me dieron la oportunidad de vivir tantas experiencias en esta vida y siempre apoyaron, aconsejaron y contribuyeron con mi desarrollo personal, gracias.

## **SUMARIO**

El presente trabajo contiene la descripción de los pasos y consideraciones para llegar a implementar una central privada por software usando el lenguaje Asterisk (software libre de alto nivel que describe rutinas de comunicación para hardware de telefonía). Al principio se da una breve introducción teórica de conceptos de telefonía analógica y digital, conceptos del sistema operativo Linux y del lenguaje Asterisk, culminando con conceptos de protocolos de comunicación de voz sobre IP en especial del protocolo SIP (Session Initial Protocol).

Al final del documento se registran datos reales y archivos de configuración de un sistema ya en funcionamiento, costos, requerimientos de hardware y tiempos para la puesta en servicio de una central con acceso primario, líneas analógicas y diversas funcionalidades como son buzón de voz, IVR (Interactive Voice Response), llamada en espera, control de llamadas de larga distancia, control de llamadas salientes y entrantes.

Este diseño es planteado como solución para una mediana empresa, empleando una arquitectura de sistema híbrido: TDM (conexión a la RTPC) y VoIP (anexos SIP), usando como plataforma una computadora (servidor con sistema operativo Centos de Linux), tarjetas controladoras de puertos de telefonía analógicas y digitales y teléfonos IP que soportan el protocolo de señalización SIP para los anexos internos.

## INDICE GENERAL

<b>PRÓLOGO</b> .....	1
<b>CAPÍTULO I.</b>	
<b>PLANTEAMIENTO DE INGENIERÍA DEL PROBLEMA</b> .....	3
1.1. Descripción del Problema.....	3
1.2. Objetivos del Trabajo.....	3
1.3. Evaluación del Problema.....	4
1.4. Limitaciones del Trabajo.....	4
1.5. Síntesis del Informe.....	5
<b>CAPITULO II.</b>	
<b>MARCO TEÓRICO CONCEPTUAL</b> .....	6
2.1. Fundamentos de la Telefonía Análoga y Digital.....	6
2.1.1. La Red de Telefonía Pública Conmutada.....	6
2.1.1.1. Circuito Local Telefónico.....	7
2.1.1.2. Circuito Local Inalámbrico.....	11
2.1.2. Red Digital de Servicios Integrados.....	11
2.1.3. Tecnologías de Transmisión.....	11
2.1.4. Conmutación Telefónica.....	13
2.2. Fundamentos del Sistema Operativo LINUX.....	14
2.2.1. Funciones de un sistema operativo.....	14
2.2.2. Sistema Operativo UNIX.....	16
2.2.3. Sistema Operativo LINUX.....	16
2.2.4. Características del sistema operativo LINUX.....	17
2.2.5. Comandos útiles a nivel administrador en LINUX.....	18
2.3. Fundamentos del Lenguaje Asterisk.....	18
2.3.1. Introducción a Asterisk.....	18
2.3.2. Evolución de Asterisk.....	20

2.3.3	Requisitos para poner en servicio un sistema de telefonía en Asterisk.....	21
2.3.3.1.	Características del Servidor.....	22
2.3.3.2.	Estabilidad y rendimiento.....	22
2.3.3.3.	Hardware de telefonía.....	23
2.3.3.3.1.	Tarjetas de telefonía análogas.....	24
2.3.3.3.2.	Tarjetas de telefonía digitales.....	24
2.3.3.4.	Conexión a redes y dispositivos IP.....	24
2.3.4.	Instalación de Asterisk.....	25
2.3.5.	Configuración de archivos principales de Asterisk.....	28
2.3.5.1.	Configuración del driver Dahdi.....	28
2.3.5.2.	Herramientas del driver Dahdi para verificación de canales.....	28
2.3.5.3.	Descripción de parámetros principales del archivo /etc/dahdi/system.conf.....	28
2.3.5.3.1.	Configuración de las troncales El acceso primario.....	28
2.3.5.3.2.	Configuración de los puertos FXS o FXO.....	29
2.3.5.3.3.	Carga de tonos.....	29
2.3.5.3.4.	Cancelación de eco.....	29
2.3.5.4.	Configuración de diferentes tipos de canales.....	29
2.3.5.4.1.	Descripción de parámetros principales del archivo chan_dahdi.conf.....	30
2.3.5.4.2.	Configuración de canales SIP.....	31
2.3.6.	Plan de Marcación.....	32
2.3.6.1.	Fundamentos del Plan de marcación.....	32
2.3.6.1.1.	Contextos.....	33
2.3.6.1.2.	Extensiones.....	33
2.3.6.1.3.	Prioridades.....	33
2.3.6.1.4.	Aplicaciones.....	34
2.3.6.1.5.	Patrones de Numeración.....	35
2.3.6.1.6.	Variables.....	35
2.4.	Protocolos de Voz sobre IP.....	35
2.4.1.	Arquitectura de protocolos de VoIP.....	35
2.4.2.	Protocolos de señalización.....	35
2.4.2.1.	H323.....	35
2.4.2.2.	SIP (Session Initial Protocol).....	37
2.4.2.3.	Diferencias entre H323 y SIP.....	40

2.4.2.4. IAX.....	40
2.4.2.5. Diferencias entre IAX y SIP.....	44
2.4.3. Protocolos de Transporte.....	44
2.4.3.1. RTP (Real-Time Transport Protocol).....	44
2.4.3.2. RTCP (Real-Time Transport Control Protocol).....	46
2.4.4. CODEC.....	47
2.4.4.1. G711 (PCMA o PCMu).....	47
2.4.4.2. G729.....	48
2.4.4.3. GSM.....	49
2.4.5. Hardware de telefonía IP.....	49
2.4.5.1 Adaptadores Telefónicos Análogos.....	49
2.4.5.2. Teléfonos IP.....	50

### **CAPITULO III**

<b>PLANTEAMIENTO DE SOLUCIÓN PARA UNA INSTALACIÓN DE ASTERISK CON 50 ABONADOS SIP Y ACCESO PRIMARIO.....</b>	<b>52</b>
3.1. Arquitectura de red.....	52
3.2. Descripción de Hardware.....	53
3.2.1. Servidor.....	53
3.2.2. Tarjeta Digium 4 FXO + 4 FXS.....	53
3.2.3. Tarjeta Digium 1 E1.....	53
3.2.4. Base celular.....	53
3.2.5. Teléfonos IP.....	54
3.3. Comparación y evaluación de costos.....	54
3.4. Cronograma de tiempos de ejecución.....	55
3.5. Direccionamiento IP.....	55
3.6. Implementación de redundancia en el servidor.....	56
3.7. Plan de Marcación.....	57
3.7.1. Numeración interna.....	57
3.7.2. Numeración externa.....	57
3.7.2.1. Llamadas entrantes.....	57
3.7.2.2. Llamadas salientes.....	58
3.8. Medidas de seguridad en la red.....	59
3.9. Puesta en servicio y configuración de los archivos del servidor Asterisk.....	60
3.9.1. Canales físicos.....	60

3.9.2. Canales lógicos y señalización.....	61
3.9.3. Usuarios y anexos SIP.....	62
3.9.4. Configurando el buzón de voz.....	64
3.9.5. Configuración del plan de marcación.....	64
3.10. Monitoreo de llamadas en el sistema Asterisk.....	68
<b>CONCLUSIONES.....</b>	<b>70</b>
<b>ANEXO A</b>	
<b>GLOSARIO DE TÉRMINOS.....</b>	<b>72</b>
<b>BIBLIOGRAFÍA.....</b>	<b>76</b>

## PROLOGO

Se es testigo del constante crecimiento económico de nuestro país a lo largo de los últimos años, en el cual una de las fuerzas impulsoras importantes son las PYMES, que aunque con escasos recursos económicos son fuentes principales de exportación e integración con las economías mundiales, así como de generación de demanda interna de bienes y servicios. Uno de ellos es el servicio de las telecomunicaciones, cuya demanda ha crecido en todo el país, ya que es el medio para establecer vínculos e intercambio económico entre personas y entes distantes.

Debido a los escasos recursos económicos, se hace necesario presentar soluciones de comunicación de costo ínfimo, gestionable por el propio soporte de la PYME, y acorde a las necesidades propias de las micro y mediana empresas; como son comunicación de larga distancia, comunicación a móviles, equipos con prestaciones inteligentes que soporten video, voz, datos y servicios suplementarios como videoconferencia, llamada en espera, buzón de voz, IVR (diálogo interactivo con el abonado llamante) a un costo cada vez más bajo.

En la actualidad existen diversas marcas que ofrecen distintas soluciones de telefonía IP como lo son Cisco con el Call Manager, Avaya, Alcatel, Mitel, etc. pero cuyas soluciones trabajan con protocolos propietarios y que difícilmente interactúan con otras plataformas.

La solución planteada que integra estas funcionalidades y a un bajo costo, es implementar una central por software con el lenguaje Asterisk administrable, usar como terminales teléfonos IP con servicios suplementarios y tarjetas de telefonía digital y analógica que realicen la integración de la red TDM con el protocolo IP.

A lo largo de los capítulos se explicará la implementación y la forma de gestionar la central IP. La solución estará aplicada a una mediana empresa de 50 trabajadores pero es válida también para una empresa de hasta 100 trabajadores, ya que se implementará sobre una red Ethernet escalable; esto teniendo en cuenta el probable crecimiento y la rápida adaptación de la red ante un aumento de tráfico.

Cabe señalar que la red Ethernet base 100 Mbps, es la común que tiene cualquier micro o mediana empresa con equipos como: PC's, hubs, switches (que no necesariamente manejan VLAN (estándar 802.1q), firewall, router gateway Internet, etc.; no hay motivo para cambiar estos equipos a menos que se requiera de una alta calidad en el servicio de voz, para lo cual se sugeriría contar con una red que soporte el estándar 802.1q y algunos otros estándares de priorización de tráfico.

Agradezco la colaboración desinteresada de mis compañeros de trabajo y de todas las fuentes de información.

## **CAPITULO I.**

### **PLANTEAMIENTO DE INGENIERÍA DEL PROBLEMA**

#### **1.1. Descripción del Problema**

El servicio de las telecomunicaciones es esencial para una pequeña y mediana empresa, la necesidad de estar comunicados, incluso en tiempo real, se hace imperante y necesaria para el intercambio económico con diversos agentes. Los servicios de voz y datos generalmente requeridos por una empresa son la telefonía fija, larga distancia, comunicación a móviles, fax, acceso a Internet, enlaces entre sucursales, etc.

Para los servicios que implican voz comúnmente se valen de una central privada llamada PBX (Private Branch Exchange) de alguna marca propietaria, generalmente de alto costo, que administra todas sus comunicaciones internas y externas; pero que simplemente es una “caja negra” con sus diversas funcionalidades, en la cual estamos limitados en la gestión y en la implementación de requerimientos propios de cada empresa, ya que el mantenimiento y configuración de esta central la realiza únicamente el soporte del proveedor de la central.

Ante esto surge la necesidad de implementar un sistema gestionable y administrable por el propio soporte de la PYME que abarate los costos para el servicio de comunicaciones de voz y contribuir con la competitividad en el mercado.

#### **1.2. Objetivos del Trabajo**

Dar una solución alternativa con un sistema de telefonía IP para las medianas empresas, que les conlleve a un ahorro económico en sus comunicaciones de voz externas.

Dar a conocer e impulsar el desarrollo del lenguaje de software libre Asterisk, lenguaje que nos permite administrar nuestras comunicaciones multimedia sin límites.

Demostrar las ventajas económicas y tecnológicas que tiene un sistema de telefonía IP ante la adquisición de una central privada con similares funcionalidades y prestaciones.

Describir y dar una idea general de todos los puntos a tener en cuenta para la implementación de un sistema de telefonía IP.

### **1.3. Evaluación del Problema**

Cuando se adquiere una central privada PBX de alguna marca propietaria, se escoge simplemente una, cuyas dimensiones y características sean las más cercanas a los requerimientos de la empresa, tales como, cuántos accesos primarios se requiere (esto va de acuerdo a la cantidad de tráfico de voz simultáneo que se tenga con la RTPC), cuantas troncales análogas son necesarias, con qué servicios suplementarios se desea contar, cuántos anexos internos soporta, etc. Cualquier aumento de capacidad o configuración compleja de la PBX tiene que ser solicitado al soporte del proveedor, lo cual involucra costos adicionales y mayores tiempos de implementación.

Ante ello existe la alternativa de implementar un sistema de telefonía IP con una central privada por software, pero se hace necesario que este sistema utilice como recurso IP, la red Ethernet ya implementada en una empresa que generalmente no contiene equipos con estándares de calidad para tráfico de voz, sino simples switches de capa dos, hubs, firewall y su respectivo router de acceso a Internet. Normalmente para una empresa pequeña de no más de 100 posiciones, los temas de calidad y capacidad no son determinantes ni limitantes debido a que generalmente una PYME tiene una sola sucursal, no hay enlaces WAN entre sucursales que puedan influir en retardo o jitter significativo para las comunicaciones de voz.

En cuanto a la capacidad o ancho de banda, una red Ethernet es de 100 Mbps y una comunicación de voz activa con codec G711 (incluyendo las cabeceras de trama TCP/IP) llega a los 88 kbps, lo cual significaría que nuestra red Ethernet tendría una capacidad para soportar más de 1000 llamadas VoIP activas internas y/o externas, dejando capacidad disponible para otro tipo de tráfico.

En cuanto al tema de abaratar los costos de comunicación, esto se logra mediante el control de las llamadas de larga distancia y a destinos móviles que se realizan desde los anexos internos, implementando restricciones, en el caso de larga distancia contratando planes especiales con el proveedor de la telefonía pública, en el caso de móviles utilizando equipos llamados “bases móviles o liceas” para aprovechar los costos reducidos que tienen las llamadas entre móviles del mismo operador.

### **1.4. Limitaciones del Trabajo**

El presente trabajo en su solución, sólo toma una de las posibles arquitecturas de telefonía IP dejando de lado otras posibles soluciones, quizás con mejor rendimiento o de

menor costo, pero cuyos resultados solo serían capaces de medirse estando implementada en una red real.

Una de las limitaciones en este trabajo es que sólo se puede escoger algunos equipos de la gama de diversas marcas y tecnologías de equipos que existen en la actualidad para telefonía IP, estos equipos fueron escogidos de acuerdo al costo y a la experiencia en uso de los mismos en otras soluciones empresariales.

Cabe señalar que para llegar a la implementación y gestión de la central en software Asterisk, es necesario que el administrador conozca conceptos básicos del sistema operativo Linux y de programación, asimismo conceptos de telefonía analógica y digital que van a ser tratados más adelante en este informe.

### **1.5. Síntesis del Informe**

A lo largo del capítulo II se expone el marco teórico y conceptual base para el desarrollo del presente informe. En primer lugar se expone acerca de los fundamentos de la telefonía analógica y digital así como se dan alcances de la RTPC.

A continuación se da a conocer conceptos acerca del sistema operativo Linux, así como de los comandos para su administración, ya que son necesarios para gestionar y configurar el servidor Asterisk que tendrá como plataforma el sistema operativo Linux distribución Centos.

Más adelante se da a conocer los fundamentos del lenguaje Asterisk, su instalación sobre Linux, la forma de configurar sus principales archivos, la configuración de los canales VoIP y TDM (FXS, FXO y RDSI) y los conceptos y herramientas en que se basa Asterisk para controlar el flujo de llamadas.

A continuación se describen los protocolos más resaltantes de la telefonía IP, en este informe se da especial énfasis al protocolo SIP ya que este será el protocolo de señalización entre los anexos internos de la empresa y el servidor Asterisk.

En el capítulo III se detalla la arquitectura de red de la solución planteada para la PYME con 50 puestos de trabajo incluyendo el sistema de telefonía IP con protocolo SIP, la conexión a la RTPC con interfaces FXS, FXO y el acceso primario, el equipamiento necesario, comparación de costos entre la solución IP y la adquisición de una central PBX, la configuración de los archivos necesarios en el servidor Asterisk incluyendo los canales físicos, lógicos y la parte principal del sistema que es el plan de marcación.

## **CAPITULO II.**

### **MARCO TEÓRICO CONCEPTUAL**

#### **2.1. Fundamentos de la Telefonía Análoga y Digital**

##### **2.1.1. La red de Telefonía Publica Conmutada**

Históricamente los sistemas telefónicos fueron diseñados sólo para transmitir señales de voz que se originaban desde una localidad distante. Hoy en día los sistemas telefónicos modernos muy sofisticados usan grandes computadoras en la oficina telefónica central para conmutar las llamadas y monitorear el rendimiento del sistema. Las oficinas telefónicas centrales enrutan voz, datos y video TDM PCM a terminales remotas distantes y otras oficinas centrales.

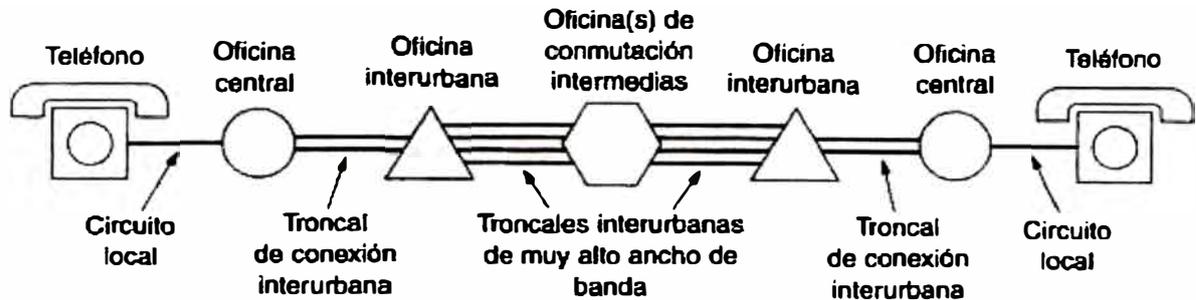
El sistema telefónico está organizado en una jerarquía de múltiples niveles con alta redundancia. Cada teléfono tiene dos alambres de cobre que van directamente a la oficina central local de la compañía telefónica, por lo general la distancia va entre 1 a 10 Km., a este conjunto del teléfono y la oficina central local se le denomina circuito local.

Si un suscriptor conectado a una oficina central llama a uno conectado a la misma, la conmutación se llevará a cabo dentro de la oficina central, por medio de un mecanismo de conmutación que establecerá una conexión eléctrica entre los dos circuitos locales y permanecerá *intacta mientras dure la llamada*.

Si el teléfono al que se llama está conectado a otra oficina central, normalmente estas oficinas centrales tienen interconexión con las centrales llamadas “interurbanas” (si están dentro de la misma localidad se les llama oficinas en tándem). Si sucede que tanto la oficina central de quien llama como de quien es llamado tienen una troncal de interconexión con la misma oficina central interurbana, la conmutación se realiza dentro de la oficina interurbana.

Existen oficinas interurbanas regionales y a su vez están interconectadas por oficinas de conmutación intermedias que procesan un gran volumen de tráfico. En la figura 2.1 se muestra una red telefónica para una comunicación de media distancia. Para los sistemas de telecomunicación se usan varios medios de transmisión, en la actualidad consisten en pares

trenzados para el cableado terminal, entre las oficinas de conmutación se usan cables coaxiales, fibra óptica y microondas.



**Figura 2.1. Ruta típica de un circuito para una llamada de media distancia**

En el pasado la transmisión en el sistema telefónico era analógica, actualmente las troncales y conmutadores son digitales y el circuito local queda como el único componente de tecnología analógica. En síntesis el sistema telefónico consta de tres componentes principales:

- a.- Circuitos locales (cables de par trenzado que van hacia el usuario)
- b.- Troncales (fibra óptica digital que conecta las oficinas de conmutación)
- c.- Oficinas de conmutación (donde las llamadas pasan de una troncal a otra)

#### **2.1.1.1. Circuito local telefónico**

El circuito local telefónico está conformado por el lazo local de abonado (local loop) y la central local office exchange. Los sistemas modernos de telefonía han evolucionado con base en la simplicidad del circuito analógico inventado por A. Graham Bell en 1876. En la figura 2.2 se muestra una gráfica simplificada del circuito telefónico.

Dos auriculares telefónicos se conectan mediante un par trenzado, los aparatos telefónicos son alimentados por una batería localizada en la Oficina central, la cual produce una corriente cd. Se usa un elemento de micrófono de carbón en el auricular, el cual consiste en granos de carbón sueltos empacados en el diafragma (caja flexible), los granos de carbón se comprimen y descomprimen, esto crea una resistencia variable que modula la corriente de cd del circuito. El lado del auricular por donde se escucha, consta de un electro magneto con un diafragma paramagnético colocado dentro de campo magnético. La corriente variable que pasa a través del electro magneto causa que vibre el diafragma del auricular y reproduce el sonido.

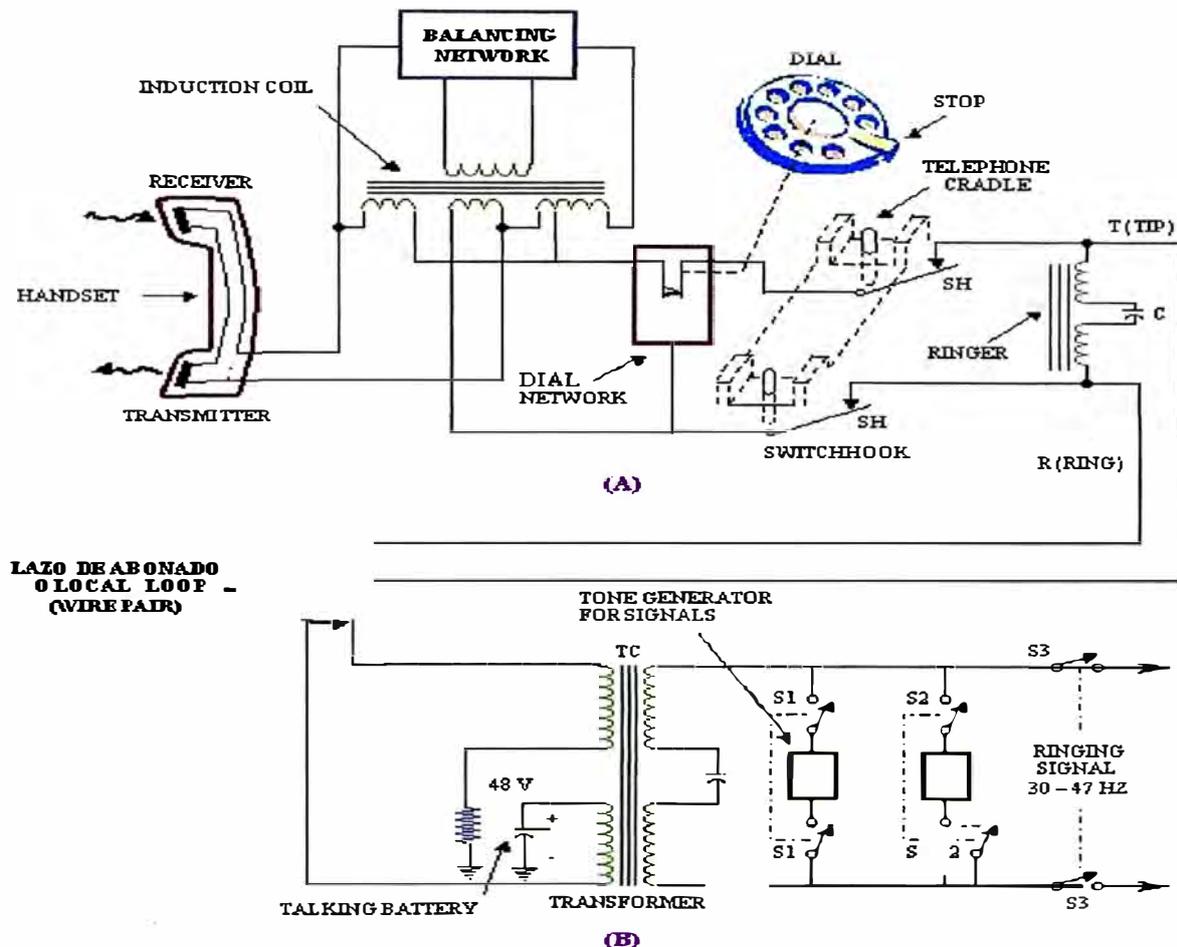


Figura 2.2. (A) Aparato telefónico (B) Central local

Las ventajas de un circuito de dos hilos son (1) no es caro, (2) los equipos telefónicos son alimentados desde la oficina central, (3) comunicación full dúplex. La desventaja es que no puede utilizar amplificadores telefónicos, por lo que para distancias grandes se usan circuitos de cuatro hilos.

La oficina de conmutación local conecta a dos lazos locales mediante acoplamiento de transformadores.

Es básicamente una conexión en serie con el audífono, el micrófono y la batería. El cable con voltaje positivo del CO se llama terminal en punta y el color rojo se llama anillo.

A continuación se describe la serie de eventos que ocurren cuando se realiza una llamada telefónica. La parte que realiza la llamada levanta el auricular, esta acción cierra el contacto del interruptor lo cual hace que fluya la corriente de cd (aprox. 40 mA) en el circuito local. La corriente se detecta en la CO y provoca que la CO proporcione un tono de marcado de 400 Hz aprox. Quien llama marca el número usando la marcación por pulsos o por tono.

Hasta que se reciba la secuencia completa la CO genera la señal de timbrado (90 Vrms, 20 Hz), esto hace sonar el teléfono, cuando contesta, la corriente de cd fluye para señalar a la CO que detenga el timbrado y conecte a las dos partes a través del interruptor del circuito. La corriente cd fluye en la línea de ambas partes, mediante el transformador de acoplamiento. Cuando habla cualquiera de las dos partes las vibraciones en el sonido causan que la resistencia del carbón cambie, así que la corriente cd de la línea se modula. Esto provoca la señal de audio ca en la otra parte. Ambas partes podrían hablarse y escucharse simultáneamente.

En resumen el usuario de un teléfono analógico está conectado a una CO o a una terminal remota a través de una línea suscriptor de par trenzado llamada local loop o lazo de abonado. Las funciones de la CO o de la TR son las siguientes:

- a.- Termina a línea con una carga de 900 ohmios (balanceada con respecto a tierra)
- b.- Proporciona corriente de circuito de cd desde su batería a través de resistores balanceados de 200 ohmios
- c.- Monitorea la corriente del circuito para determinar las condiciones de colgado y descolgado
- d.- Recibe información de marcado desde y hacia el usuario vía marcación de tonos y pulsos
- e.- Aplica el voltaje de timbre y la señal de llamada en progreso (tono de marcación, señal de ocupado, tono de llamada, etc.)
- f.- Convierte una línea de dos hilos a una de cuatro para usarse en transmisión y recepción simultáneamente
- g.- Proporciona conversión analógica a digital y digital a analógica para sistemas digitales a través de PCM
- h.- Prueba la línea del suscriptor y la terminación de la línea
- i.- Proporciona aislamiento de protección del equipo CO/TR de potenciales de línea indeseados como rayos o conexiones cruzadas de corriente alterna.
- j.- Contiene equipo de conmutación, de señalización y baterías de corriente directa.

La CO responde ante los números marcados por tonos o pulsos y establece la conexión entre los lazos de abonado. La función de señalización se establece de la siguiente manera:

- **Iniciando una llamada:**

**On Hook:** señal de colgado, circuito abierto, circuito de timbrado abierto. Impedancia grande (20 – 25 Mohm), corriente nula (0.02 mA), voltaje (-48 V)

**Off Hook:** señal de descolgado, circuito cerrado con la CO, impedancia pequeña, hay flujo de corriente y la CO envía el tono de invitación al marcado. Impedancia (250 ohm), corriente (20 mA), voltaje (-5 a -12 V).

**Marcación de un número:**

**Marcación por pulsos:** abre y cierra el lazo de abonado a frecuencia de 5 Hz (descontinuado)

**Marcación por tonos:** teléfono y central equipada para enviar tonos DTMF. A continuación se muestra el grupo de frecuencias en la tabla 2.1.

**Tabla 2.1. Combinación de frecuencias para tonos DTMF**

Botón del teclado	Grupo de Frecuencias Altas	Grupo de Frecuencias Bajas
1	1209	697
2	1336	697
3	1477	697
4	1209	770
5	1336	770
6	1477	770
7	1209	852
8	1336	852
9	1477	852
*	1209	941
0	1336	941
#	1477	941
A	1633	697
B	1633	770
C	1633	852
D	1633	941

- **Realizando la conexión:**

Si el abonado está ocupado se le envía el tono de ocupado al abonado llamante. Si el teléfono está disponible se le envía al abonado llamante el tono de ring back y al abonado llamado se le envía la señal de timbrado, cuando el abonado contesta se cierra el lazo.

La señal de timbrado es una señal de CA de frecuencia cercana a los 20 Hz con un voltaje entre los 80 y 130 Vrms, montada sobre los -48 Vdc de la batería, el tiempo de encendido es de 1 a 1.5 segundos y de apagado 4 a 4.5 segundos.

#### **2.1.1.2. Circuito Local Inalámbrico**

Ante la dificultad para realizar tendido de cable o fibra, sobre todo para los permisos municipales y para llegar a lugares lejanos, las empresas operadoras de telefonía fija han encontrado una alternativa menos costosa como es el circuito local inalámbrico WLL (Wireless Local Loop) en lugar del tradicional circuito de par trenzado. Las características de estos circuitos son las siguientes: también se brinda acceso a Internet con velocidades parecidas a una línea suscriptora ADSL, las antenas direccionales son instaladas y de un tamaño mediano, el usuario no se mueve diferenciándose del servicio móvil.

En el Perú tenemos a varios operadores como Americatel, Claro y Movistar quienes han incursionado en la telefonía fija inalámbrica en los últimos años en diferentes bandas como lo son la de los 800 Mhz y el Wimax a 2 y 3 Ghz.

#### **2.1.2. Red Digital de Servicios Integrados**

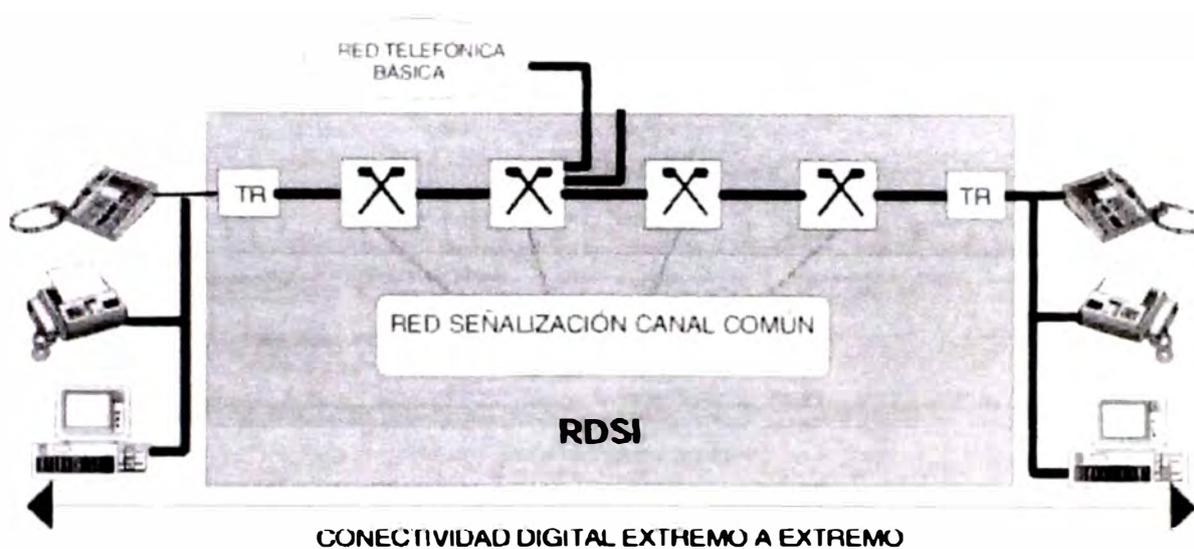
En la actualidad las CO son digitales y proporcionan salida digital a las TR, la última milla desde la TR hacia el usuario usualmente es analógica, pero ahora con la red digital de servicios integrados se convierte la última milla en una línea suscriptora digital (DSL) así que los datos digitales pueden entregarse directamente desde el usuario, el usuario puede desmulticanalizar para proporcionar distintas aplicaciones simultáneamente (1) decodificar datos para generar señales de voz, (2) decodificar datos para video, (3) procesar datos de telemetría y/o aplicaciones de PC.

Existen dos categorías de RDSI de banda angosta y la de banda ancha o acceso primario. En América del Sur y Europa se usa la trama de 2048 Mbps que se traduce en 32 tramas de 8 bits en un tiempo de 125 useg. Consta de 32 canales 2 canales tipo D y 30 canales tipo B. Los canales D transportan la señalización y los canales B los datos. En la figura 2.3 se muestra un esquema de la RDSI.

#### **2.1.3. Tecnologías de transmisión**

La tecnología WDM es excelente, pero aún ha mucho cable de cobre en el sistema telefónico que hay que aprovechar, TDM puede manejarse por completo con dispositivos digitales. Puesto que los circuitos locales producen señales analógicas se necesita una

conversión de señales analógicas a digitales en las oficinas de centrales de conmutación en donde todos los circuitos locales se combinan y forman troncales. Las señales analógicas se digitalizan en la oficina central con un dispositivo llamado codec (codificador - decodificador), con lo que se produce una serie de números de 8 bits. El codec toma 8000 muestras por segundo (125useg/muestra) porque el teorema de Nyquist dice que es suficiente para capturar toda la información del ancho de banda de 4 KHz. del canal telefónico; a una velocidad menor se perdería información; a una mayor, no se ganaría información extra.



**Figura 2.3. Esquema de la RDSI**

Esta técnica se llama PCM (Modulación por codificación de Pulsos). La PCM es el corazón del sistema telefónico moderno, en consecuencia todos los intervalos de tiempo dentro del sistema telefónico son múltiplos de 125 useg. El CCITT recomienda una trama de 2048 Mbps llamada E1. Esta trama empaqueta 32 muestras de datos de 8 bits en la trama básica de 125 useg, treinta de los canales se usan para información y dos para señalización. Cada grupo de cuatro tramas proporciona 64 bits de señalización, la mitad de los cuales se reserva para señalización por canal asociado, el resto se usa para sincronización de tramas. En forma práctica, el primer canal es utilizado para sincronismo, los siguientes quince canales para datos, el canal dieciséis para señalización y los últimos 15 canales para datos. A continuación se muestra una representación de la trama E1 en la figura 2.4.

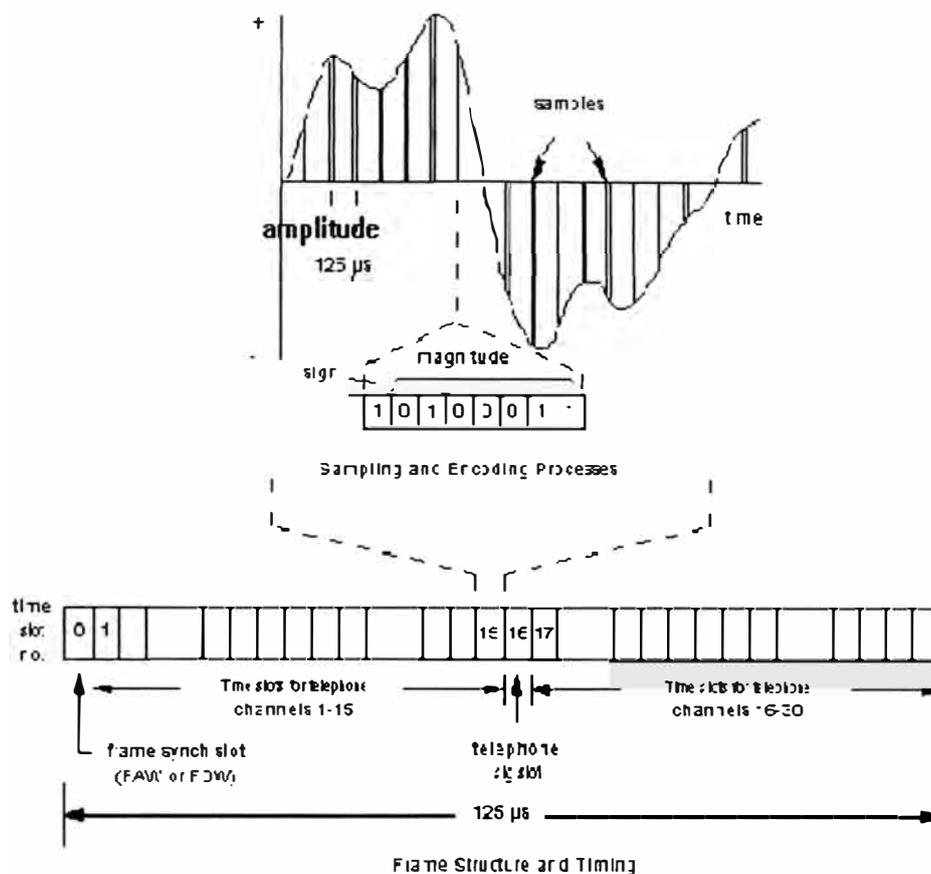


Figura 2.4. Trama E1

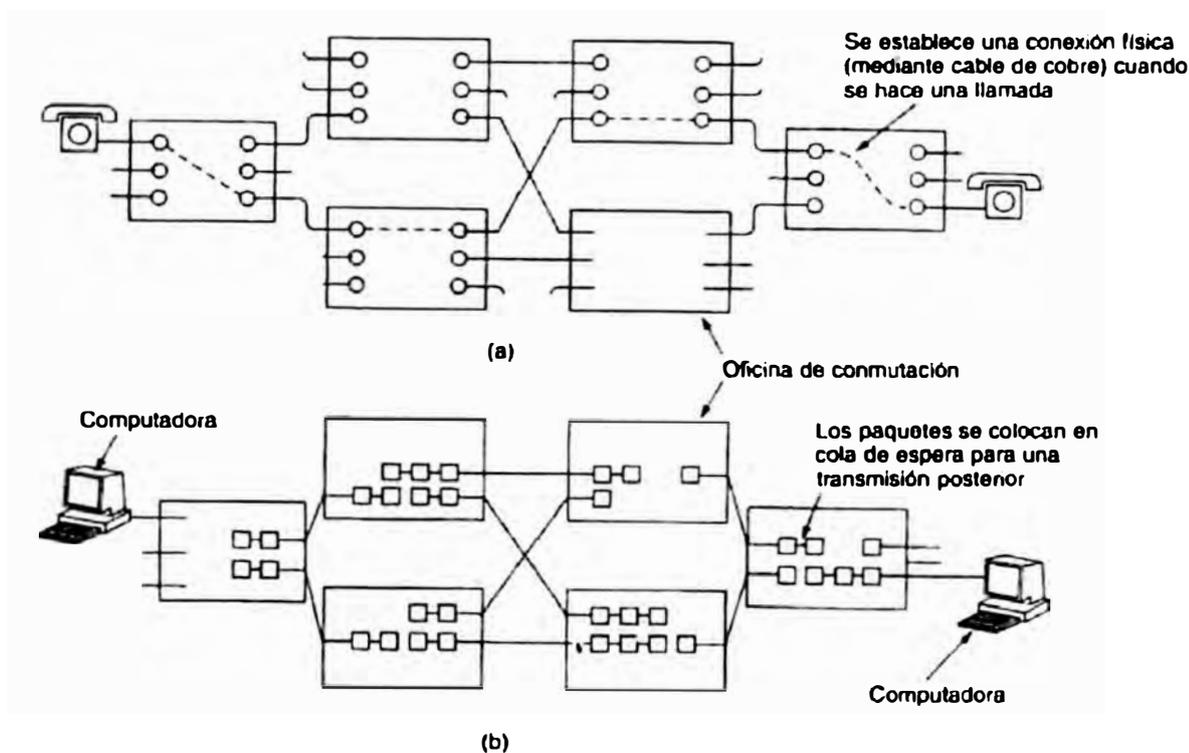
#### 2.1.4. Conmutación Telefónica

Desde el punto de vista ordinario de un ingeniero telefónico, el sistema se divide en dos partes: planta externa (circuitos locales y troncales) y planta interna (los conmutadores que están dentro de las oficinas de conmutación).

En la actualidad se utiliza dos técnicas diferentes de conmutación: conmutación de paquetes y conmutación de circuitos. La técnica usada por el sistema de telefonía es la de circuitos. Cuando alguien origina una llamada, el equipo de conmutación busca una trayectoria física que va a desde su teléfono al del receptor, la representación de cómo se establecen los caminos lógicos y físicos se muestra en la figura 2.5.

Cada uno de los seis rectángulos representa una oficina de conmutación (central, interurbana, etc.). Cuando una llamada pasa por una oficina de conmutación se establece una conexión física (conceptual) entre la línea por la que llegó la llamada y una de las líneas de salida. Este modelo está altamente simplificado porque partes de la trayectoria

pueden ser de hecho, enlaces microondas o fibra en los cuales se multiplexan miles de llamadas. Sin embargo la idea es válida: una vez que se ha establecido una llamada existe una trayectoria dedicada entre ambos extremos y continuará existiendo hasta la finalización de la llamada.



**Figura 2.5. a) Conmutación de circuitos b) Conmutación de paquetes**

Una propiedad de la conmutación de circuitos es la necesidad de establecer una trayectoria de un extremo a otro antes de que se pueda enviar cualquier dato. Una ventaja de la trayectoria establecida es que no hay peligro de congestión, es decir una vez que la llamada toma un circuito no hay posibilidad de obtener una señal de ocupado, aunque si se pudiese obtener antes de establecer la conexión debido a falta de capacidad de conmutación de troncal.

## 2.2. Fundamentos del Sistema Operativo LINUX

### 2.2.1. Funciones de un sistema operativo

El sistema operativo reside en el disco duro de la computadora y actúa como un puente entre los programas de usuario y los programas que controlan el hardware de la computadora.

El sistema operativo trabaja principalmente en segundo plano. Se encarga de los componentes hardware de una computadora, además de asegurar el inicio y ejecución de diversos programas. También se encarga de los datos almacenados en los dispositivos de almacenamiento de la computadora.

El sistema operativo es el responsable de las siguientes funciones del sistema de computadora:

- Arrancar o iniciar la computadora
- Actuar como interfaz entre el CPU y el mundo externo
- Coordinar los dispositivos del sistema
- Coordinar las aplicaciones o programas en ejecución
- Proporciona diferentes recursos a los usuarios, tales como: cálculo, almacenamiento, dispositivos de Entrada/Salida (Input/Output – I/O), manejo de red, etc.
- Permite que varios usuarios trabajen juntos compartiendo e intercambiando programas, aplicaciones y datos en la misma instalación.
- Ayuda a resolver conflictos cuando los usuarios solicitan el mismo recurso simultáneamente.
- Proporciona seguridad cuando los usuarios comparten datos y programas.
- Asiste en la administración y evaluación del uso y eficacia de un sistema, recolectando datos sobre la utilización de los recursos.

Los sistemas operativos son clasificados en dos tipos:

- **Sistema Operativo de Usuario Único:** permite que solo un usuario use todos los recursos del sistema de computadora en cualquier momento dado. Mientras el procesador maneja un usuario o programa, otros programas o usuarios no pueden usar estos recursos. Un ejemplo de este tipo el sistema operativo DOS.
- **Sistema Operativo Multiusuario:** permite que más de un usuario o programa se ejecuten o usen recursos del sistema en forma simultánea. Algunos ejemplos de estos sistemas operativos multiusuario son UNIX, Linux, Solaris, etc. En los tiempos en que los sistemas operativos como DOS únicamente permitían solo un usuario para realizar tareas simples la introducción de UNIX creó una nueva forma de computación. A pesar que Windows también es un sistema operativo principalmente de usuario único, permite que varias tareas se realicen en forma simultánea.

### 2.2.2. Sistema Operativo UNIX

Es conocido como un sistema operativo abierto dado que puede ser llevado e instalado en cualquier clase de sistema de computadora y plataforma de hardware. Normalmente, los sistemas operativos son escritos en lenguaje ensamblador. Sin embargo, UNIX está escrito en un lenguaje de alto nivel y su código fuente está disponible fácilmente. Esto supone una ventaja para los programadores cuando incorporan cambios para ajustarse a sus necesidades.

Es un sistema multiusuario y multitarea. Multitarea significa que el sistema operativo coordina múltiples tareas o trabajos en forma simultánea. Mientras un usuario esta compilando un programa en C, otro puede crear documentos usando un editor, cada uno ignorando la presencia del otro.

UNIX es uno de los sistemas operativos más poderosos existentes, por el hecho de poder compartir recursos en tiempo real. A pesar de que UNIX está desarrollado para programadores, proporciona un entorno tan flexible que también es usado en negocios, ciencias, educación e industria.

### 2.2.3. Sistema Operativo LINUX

Linux es un sistema operativo distribuido gratuitamente basado en el sistema operativo UNIX. Fue desarrollado originalmente por Linux Torvalds, quien empezó a trabajar sobre Linux en 1991 siendo estudiante de la Universidad de Helsinki en Finlandia. Luego, miles de programadores contribuyeron a su desarrollo y fue distribuido gratuitamente por Internet.

En 1990, la mayoría de las piezas de software del sistema operativo basado en GNU se habían escrito, excepto la más importante, el kernel. Más tarde, el kernel gratuito basado en UNIX, desarrollado por Linux Torvalds fue combinado con el sistema GNU. Así nació un sistema operativo, el sistema GNU basado en Linux.

Las etapas significativas en la evolución de Linux son:

- En 1991, Linux Torvalds desarrolló Linux con el soporte de desarrolladores a lo largo del mundo y lo llamó Linux
- Él lanzó la versión 0.02 de Linux en 1991
- En 1994, fue lanzada la versión 1.0 de Linux
- La versión 2.6 actual, completa fue lanzada en Diciembre de 2003. Sin embargo su desarrollo continúa.

El sistema operativo Linux está organizado funcionalmente en los siguientes tres niveles:

- Kernel (Núcleo)
- Shell (Intérprete de Comandos)
- Herramientas y aplicaciones

El kernel es el núcleo de un sistema operativo, así como la CPU es el núcleo de un sistema de computadora. El kernel es una colección de programas, la mayoría escritos en C y solo existe un kernel para cualquier sistema operativo. Este se comunica directamente con el hardware y sin un kernel, un sistema operativo no puede interactuar con el hardware.

El sistema operativo Linux usa un shell para transferir los comandos desde el teclado a la computadora. El shell (intérprete de comandos) es solo otro programa escrito en C. Actúa como un intérprete entre los programas de los usuarios y el kernel. Traduce los comandos del usuario en la acción apropiada. El shell interactúa con el usuario, mientras que el kernel interactúa con el hardware de la máquina. El shell es el programa que toma comandos y ejecuta el programa apropiado o lo traduce.

En el sistema Linux existe un cierto número de herramientas disponibles. Las herramientas son programas de usuario que pueden ser escritos por terceros para determinados tipos de aplicaciones. Típicamente, las herramientas se agrupan para realizar ciertas funciones, tales como programación, aplicaciones de negocio y procesamiento de texto.

#### **2.2.4. Característica del sistema operativo LINUX**

El sistema Linux ofrece las siguientes características:

- Estabilidad: tiene protección de la memoria entre procesos, de manera que uno de ellos no pueda colgar el sistema.
- Multitarea: varios programas (realmente procesos) ejecutándose al mismo tiempo
- Multiplataforma: se ejecuta en muchos tipos de CPU, no solo Intel
- Multiusuario: varios usuarios en la misma máquina al mismo tiempo y sin licencias para todos
- Manejo de la Memoria: la memoria se gestiona como un recurso unificado para los programas de usuario y para cache de disco, de tal forma que toda la memoria libre puede ser usada para cache y este puede a su vez ser reducido cuando se ejecutan grandes programas

- Interfaz Gráfica de Usuario: KDE, GNOME.
- Desarrollo de Software: KDevelop (Lenguaje C, C++, Java, PHP, Perl, Python, entre otros)
- Trabajo de redes: TCP/IP, incluyendo ftp, telnet, NFS, Wireless, etc.
- Disponibilidad del código Fuente: todo el código fuente está disponible, incluyendo el núcleo completo, todos los drivers, las herramientas de desarrollo y todos los programas de usuario.

### 2.2.5. Comandos útiles a nivel administrador en LINUX

La forma de ingresar comandos en Linux es parecido al del sistema operativo DOS. Es decir una línea de comando tiene tres partes principales:

- Comando: Indica al shell la acción a realizar
- Opciones: Se indica las modificaciones al comando
- Argumento: Caracteres (archivo o directorio) que es la entrada al comando

Para obtener una descripción amplia de algún comando se utiliza el comando “man”

Ejemplos:

```
root$ tar -xvf archivo.bin
```

```
root$ man tar
```

Se menciona algunos comandos comunes y útiles en la tabla 2.2.

## 2.3. Fundamentos del Lenguaje Asterisk

### 2.3.1. Introducción a Asterisk

Asterisk es un completo PBX (Private Branch Exchange) por software para múltiples plataformas bajo los sistemas operativos LINUX, FreeBSD y otros; donde las llamadas en el sistema ejecutan funciones a través de patrones de dígitos (mejor conocidos como extensiones), ofreciendo un completo control sobre el enrutamiento de las mismas con relativa facilidad.

Incluye funcionalidades encontradas en los sistemas de conferencias, audio respuesta interactiva (Interactive Voice Response), música en espera y otras funcionalidades más avanzadas que permiten la interconexión con sistemas de telefonía externos a través de troncales analógicas, digitales o las más avanzadas opciones con interfaces para VoIP tales como SIP, H323, IAX entre otros más; no sólo para comunicaciones de voz sino también para video.

Tabla 2.2. Comandos útiles y comunes en LINUX

COMANDO	ACCIÓN
<b>adduser</b>	Añade usuarios al sistema.
<b>cat</b>	Muestra todo el contenido de un fichero sin pausa alguna
<b>chmod</b>	Cambia los permisos de lectura/escritura/ejecución de ficheros/directorios
<b>chown</b>	Cambia los permisos de usuario, grupo de ficheros o directorios
<b>cp</b>	Copia ficheros/directorios
<b>date</b>	Gestión de fecha de sistema
<b>df</b>	Muestra la ocupación del disco duro
<b>free</b>	Muestra el estado de la memoria RAM y el SWAP
<b>grep</b>	Filtra los contenidos de un fichero.
<b>gunzip</b>	Descompresor compatible con zip
<b>head</b>	Muestra la cabecera o 10 líneas iniciales de un fichero
<b>history</b>	Muestra el historial de comandos introducidos por el usuario
<b>ifconfig</b>	Muestra la configuración de interfaces de red
<b>ln</b>	Crea enlaces, accesos directos
<b>ls</b>	Lista contenido de directorios
<b>mkdir</b>	Crea directorios
<b>modprobe</b>	Trata de instalar un módulo, si lo encuentra lo instala pero de forma temporal
<b>more</b>	Muestra el contenido de un fichero con pausas cada 25 líneas
<b>mount</b>	Monta unidades de disco duro, diskette, cdrom
<b>mv</b>	Renombra o mueve ficheros/directorios
<b>netstat</b>	Muestra las conexiones de red actuales
<b>ps</b>	Muestra información sobre procesos en ejecución
<b>rm</b>	Elimina ficheros/directorios
<b>rmdir</b>	Elimina directorios
<b>rpm</b>	Gestor de paquetes de redhat. Para instalar o actualizar software de sistema
<b>sort</b>	Ordena el contenido de un fichero
<b>tail</b>	Muestra 10 líneas de un fichero
<b>tar</b>	Archivador de ficheros
<b>umount</b>	Desmonta unidades
<b>uname</b>	Indica la versión de kernel, procesador, y nombre de sistema
<b>userdel</b>	Elimina usuarios del sistema
<b>usermod</b>	Modifica usuarios del sistema
<b>wget</b>	Programa para descargar ficheros por http o ftp
<b>who</b>	Muestra los usuarios de sistema que tienen sesión activa
<b>whois</b>	Consulta de dominios

Esta poderosa combinación de funcionalidades permite construir aplicaciones tan complejas o avanzadas como se desee, sin incurrir en elevados costos y con más flexibilidad que en cualquier sistema de telefonía.

Actualmente Asterisk se usa en Call Centers, Contact Centers, reemplazo de PBX, Gateway de VoIP para terminación de voz, etc.

### **2.3.2. Evolución de Asterisk**

En la década anterior a Asterisk, los grandes productores de soluciones de telecomunicaciones dominaban el poder y mantenían al margen del desarrollo de la telefonía a los libres desarrolladores. Así por ejemplo una central PBX como la Alcatel 4400 basada en un viejo sistema Unix AT&T con más de 20 años de antigüedad, y unas aparentemente poderosas CPU's basadas en viejos procesadores Intel 386 que nos permitían manejar puertos análogos, digitales, primarios, música en espera grabada en memorias flash, todo por una suma exorbitante de dólares, sólo por la CPU, algunas tarjetas y el antiguo sistema operativo licenciado. Otro punto era que los teléfonos digitales eran propietarios y no se podía conectar algún teléfono de otra marca.

El paradigma de que las tarjetas de telecomunicaciones deberían tener un enorme juego de chips de procesamiento de señales (DSP), hacía costosa la construcción de tarjetas de este tipo. Pero un día a alguien se le ocurrió hacer que las tarjetas prescindan del DSP y hacer que las nuevas y poderosas CPU's se hagan cargo del procesamiento.

Jim Dixon creía firmemente que las nuevas CPU's basadas en el x86 de Intel tenían el poder necesario para proveer las funcionalidades DSP, y el bajo costo de estas CPU's harían más atractiva la solución que los costosos juegos de chips DSP usualmente embebidos en las tarjetas de telefonía.

Dixon como otros visionarios, pensó que su pensamiento era bastante obvio y muchos otros seguirían el mismo camino, pero al cabo de algunos años vio que aún nadie creaba tarjetas como aquellas así que decidió iniciar la revolución el mismo.

Sus primeros trabajos fueron con una tarjeta Mitel MB89000C "ISDN Express De elopment" que ofrecía un poco de todo a los desarrolladores de telefonía y contenía un par de interfaces T1 y una matriz cruzada que le daba acceso desde el bus ISA a los timeslots del T1.

El siguiente paso de Dixon fue escribir un driver para controlar esta tarjeta y esta vez lo hizo para FreeBSD, el sistema operativo con el que trabajaba en el momento, labor que

llevo a cabo en su Pentium III de 600 Mhz. Por más increíble que parezca, el éxito de este driver (driver “mie” como solía llamarle) lo llevo a crear un nuevo diseño de tarjeta ISA que fuera más eficiente, consiguiendo manejar 2 interfaces T1 sin problemas desde su CPU Pentium III.

Conociendo lo revolucionario de su trabajo, Dixon decidió asociarlo al nombre principal de la revolución mexicana Emiliano Zapata y de allí surgió el nombre para su proyecto Zapata Telephony y su primera tarjeta fue llamada “Tormenta”.

Luego que Dixon escribiera un driver en Linux y lo publicara en la red, un joven universitario de Alabama llamado Mark Spencer se ofreció a rescribir el driver de Linux a fin de perfeccionarlo, sobre todo porque tenía un proyecto en el que el trabajo de Dixon encajaba a la perfección, algo llamado Asterisk.

En ese momento Asterisk era un concepto funcional pero inviable en la práctica ya que no contaba con una forma de conectarse con el hardware de telefonía alguno y la revolución de Zapata Telephony sería quien le diera esa capacidad.

De allí en adelante Mark Spencer y Jim Dixon trabajaron juntos y la tarjeta Tormenta ISA vio la luz en una nueva versión, la Tormenta 2 PCI con 4E1/T1, mejor conocida como las Digium E400P y T400P comercializada por la empresa de Mark Spencer.

Juntos crean el modulo chan\_zap y DIGIUM, en 2004 se lanza Asterisk 1.0, en el 2005 Asterisk 1.2 y en el 2006 el Asterisk 1.4, actualmente estamos en la versión 1.6 lanzada en el 2008.

### **2.3.3. Requisitos para poner en servicio un sistema de telefonía en Asterisk**

Primero que todo se debe decidir a que nivel se está interesado en Asterisk, como hobby, trabajo intelectual o para ponerlo en producción en una empresa.

Por ejemplo un Pentium III de 600 Mhz y 128 Mb de RAM sería una plataforma suficiente para un sistema de pruebas que soporte hasta 5 llamadas simultáneas y donde se pueda explorar todas las funcionalidades de Asterisk, mientras que como sistema operativo podriamos optar por una distribución de Linux como Fedora. Ahora bien, para una plataforma como esta podemos pensar en una tarjeta de telefonía de hasta 4 canales ZAP, por ejemplo una tarjeta Digium TDM04B. Un sistema con procesador Celeron de 1 Ghz con 256 Mb de RAM sería una plataforma suficiente para un sistemas de pequeña oficina que podria soportar de 5 a 10 llamadas simultáneas con una tarjeta de telefonía Digium TDM808B de 8 canales, una distribución de Linux con el kernel 2.6

Para un sistema mayor que pueda soportar de 10 a 24 llamadas simultáneas lo ideal es un sistema Pentium 4 de 3 Ghz con 512 Mb de RAM o los más recientes Pentium Dual Core o Quad Core desde 1.8 Ghz y una tarjeta que maneje hasta 24 canales como la Digium TDM2406B.

Para sistemas superiores a 30 llamadas simultáneas lo más recomendable es soportarse sobre sistemas Xeon con opción de segunda de CPU, 1 Gb de memoria RAM con tarjetas como la Digium TE120P para un E1/T1, o hasta la TE410P o similares para cuatro E1/T1.

### **2.3.3.1 Características del Servidor**

La selección del servidor es simple pero a la vez compleja, simple porque cualquier plataforma x86 es compatible, pero compleja ya que la estabilidad y capacidad total del sistema dependerá de múltiples factores, así será crucial la elección de la CPU, motherboard, memoria, discos duros, interfaces de red, ventilación y fuentes de poder, además de consideraciones en las opciones de ahorro de energía del sistema.

Las tarjetas Digium para bus PCI son particularmente sensibles al uso compartido de las IRQs, aunque ahora las de slot PCI Express ya no tienen esta debilidad. Aparte hay otros fabricantes de tarjetas de telefonía para Asterisk que tampoco tienen este problema con las IRQs, por ejemplo Sangoma.

De la experiencia y en la actualidad, un CPU Pentium IV Celeron soporta 30 llamadas simultáneas, un Dual Core 60 llamadas simultáneas y un CPU Quad Core hasta 120 llamadas simultáneas.

### **2.3.3.2 Estabilidad y Rendimiento**

No sólo basta con escoger un servidor poderoso, también hay que tener en cuenta que hará Asterisk sobre el servidor y como lo hará.

- Llamadas o conexiones simultáneas, dependiendo del número de conexiones simultáneas se usará más o menos capacidad del sistema.
- Cancelación de Eco, según el hardware utilizado se requerirá efectuar la cancelación de eco desde la tarjeta de telefonía o a través del software, lo que impone una carga extra a la CPU del servidor.
- Tipo de codec : El peso impuesto a la CPU por el manejo del DSP depende en gran medida del tipo de codec a usar de la necesidad o no de hacer traducciones entre estos, por ejemplo un sistema que cubra nuestras necesidades bajo G711 puede solo proveernos el 20% de capacidad usando G729.

- Servicios especiales: El tipo de servicio a utilizar afectará en gran medida el rendimiento de nuestro servidor, por ejemplo la grabación de llamadas puede absorber gran parte de la capacidad de nuestra CPU.
- Aplicaciones especiales: El uso de la lógica en los archivos de configuración de Asterisk puede requerir poco o mucho de nuestra CPU según como se haya construido la lógica del plan de marcación y del plan de numeración, más allá de eso, el uso de aplicaciones más pesadas como los AGI pueden imponer serias cargas en la CPU.
- Sistema operativo: Aunque es sabido que Asterisk puede ejecutarse en diversos sistemas operativos, se recomienda instalar Asterisk en alguna distribución de Linux con kernel 2.6
- Otros procesos: Si además de Asterisk tenemos otros procesos en el mismo sistema tales como: servidor web, bases de datos, aplicaciones en PIIP, CRM, crontabs, se debe contemplar el uso que harán estos de la CPU y la prioridad con la que se deberían ejecutar a fin de minimizar el efecto en todo el sistema.
- En el caso de la RAM, Asterisk consume cerca a 30 Mbytes.
- En el caso del disco duro, preocuparse sólo si se van a realizar grabaciones de voz, realizar mantenimiento y verificar espacio libre, es recomendable tener redundancia en cuanto a la capacidad.

### **2.3.3.3. Hardware de telefonía**

Si la solución basada en Asterisk está conectada a la RTPC, entonces se necesitará equipos telefónicos digitales y/o analógicos. Asterisk cuenta con un número creciente de equipos y tarjetas de telefonía compatibles, pudiendo conectar el servidor Asterisk a cualquier interfaz telefónica estándar en la actualidad. Luego se mencionarán algunos modelos analógicos y digitales de los fabricantes de tarjetas compatibles con Asterisk.

#### **2.3.3.3.1 Tarjetas de telefonía analógicas**

Para soluciones pequeñas y medianas con Asterisk, por lo general se utiliza conexiones analógicas a la RTPC, a sea en forma de puertos FXS (Foreign Exchange Station) o FXO (Foreign Exchange Office).

Digium, uno de los proveedores desarrolladores originales de Asterisk cuenta con la tarjetas analógicas donde la nomenclatura es la siguiente: TDM24XYB o E donde X e

numero de módulos FXS, Y número de módulos FXO, B significa que está compuesta por módulos, y E cuando la tarjeta incluye cancelación de eco.

Otra compañía que produce tarjetas análogas compatibles con Asterisk es la canadiense SANGOMA con sus líneas A200 y A400 que pueden manejar 24 y 48 puertos análogos respectivamente, con su tecnología denominada REMORA, que usa un backplane que puede unir varias tarjetas a un único bus PCI o PCI Express.

Para que las tarjetas Sangoma aparezcan como interfaces Dahdi para Asterisk, se requiere además de los drivers Wanpipe propios de Sangoma y parches de código que modifica el driver Dahdi original.

#### **2.3.3.3.2 Tarjetas de telefonía digitales**

Cuando se necesita más de 24 o 48 líneas o se requiere conectividad digital en vez de analógica, es necesario optar por tarjetas digitales para conectar puertos E1 (estándar utilizado en América del Sur y Europa) o T1 (estándar usado en USA).

En este caso Digium tiene una línea de tarjetas de las combinaciones de 1, 4 y 8 E1s, diferenciadas por el voltaje PCI, algunas para PCI Express, algunas incluyen o no, cancelación de eco que libera la carga del CPU, al evitar que Asterisk se encargue de esto y mejora la calidad de las llamadas. Adicionalmente se menciona algunos otros fabricantes de tarjeta compatibles con Asterisk: ATCOM de China, BROADTEL de China, DIGIVOICE de Brasil, OPENVOX de China, PARABEL de Rusia, VOICETRONIX de Australia.

#### **2.3.3.4. Conexión a redes y dispositivos IP**

Si no se requiere conectar Asterisk a la RTPC entonces Asterisk provee todo lo que se necesita para cubrir las necesidades, para esto Asterisk utiliza el hardware Dahdi como fuente de temporización y en caso de no tenerlo se requiere el modulo de kernel ztdummy. También se requiere de una tarjeta de red para conectar Asterisk a nuestra red y a cualquier dispositivo IP, es recomendable escoger una que imprima menor carga al CPU y con una interfaz de mayor ancho de banda (10/100/1000).

En el caso de los teléfonos IP, las opciones son cada vez más amplias y las características de los mismos están cambiando a una velocidad sorprendente donde la característica es un dispositivo que incluye voz, datos y video. Entre ellos se puede mencionar a los teléfonos fabricados por Grandstream que impulsan el concepto de bajo costo, cuya calidad es cada vez mayor con su línea de teléfonos SIP que incluye video,

aunque también se puede encontrar fabricantes con productos de perfil superior como Cisco y la alemana Snom.

También se puede utilizar Asterisk con teléfonos virtuales o SoftPhones que son programas que proveen funcionalidad telefónica en un computador, PDA o laptop. Otra alternativa es utilizar un adaptador telefónico, también conocido en el mercado como ATA (Analogue Telephone Adaptor), que permite conectar dispositivos análogos a Asterisk como si estos fueran dispositivos IP. Estos adaptadores son realmente gateways.

En el caso de las comunicaciones a móviles, existen los dispositivos llamados liceas o bases celulares que son gateways GSM a FXS o GSM a SIP.

#### **2.3.4. Instalación de Asterisk**

Asterisk requiere tres paquetes, los cuales son: Asterisk, Dahdi (anteriormente denominado zaptel) y Libpri. Asterisk es el programa principal, Dahdi incluye los drivers de telefonía para controlar las tarjetas análogas y/o digitales y el Libpri incluye las librerías necesarias para conectarse a la RTPC utilizando el protocolo RDSI (interfaces T1/E1). Adicionalmente el paquete Addons contiene librerías externas como MySQL, MP3, etc.

A continuación se muestran los pasos a seguir para la instalación de Asterisk en un Linux distribución Centos que es el preferido como plataforma. Las versiones actuales y archivos necesarios se encuentran en la página web [www.asterisk.org](http://www.asterisk.org) y la descarga es gratuita:

- Asterisk 1.6.1.0
- DAHDI Linux 2.1.0.4
- DAHDI Tools 2.1.0.2
- Libpri 1.4.9
- Addons 1.6.1.0

Adicionalmente es necesario instalar algunos paquetes requeridos por Linux, tales como librerías en C, paquetes necesarios para Asterisk, paquetes necesarios para Dahdi, archivos de sonido para el IVR en diferentes codecs e idiomas, etc.

**a. El primer paso es descargar los paquetes indicados y copiarlos en la carpeta**

**`/usr/src/asterisk`**

`[root@asterisk ~]# ls -l`

asterisk-1.6.0.9.tar

asterisk-addons-1.6.1.0.tar

dahdi-linux-2.1.0.4.tar

dahdi-tools-2.1.0.2.tar

libpri-1.4.9.tar

- b. Averiguar la versión exacta de nuestro kernel con el comando `uname -a`, se tendrá una respuesta del siguiente modo, notar la versión de kernel**

```
[root@asterisk ~]# uname -a
```

```
[root@asterisk ~]# Linux asterisk 2.6.18-8.el5 #1 SMP Thu May 15 19:57:35
EDT 2009 i686 i686 i386 GNU/Linux
```

- c. Luego verificar si se tiene instaladas las fuentes del kernel, para ello utilizamos el comando `rpm -q kernel-devel`, se debería ver algo como esto**

```
[root@asterisk ~]# rpm -q kernel-devel
```

```
kernel-devel-2.6.18-8.el5
```

Si no están instaladas, hacerlo con el comando yum de la siguiente forma:

```
[root@asterisk ~]# yum install kernel-devel
```

- d. Luego instalar los diferentes paquetes requeridos con los siguientes comandos**

```
[root@asterisk ~]# yum install bison
```

```
[root@asterisk ~]# yum install bison-devel
```

```
[root@asterisk ~]# yum install ncurses
```

```
[root@asterisk ~]# yum install ncurses-devel
```

```
[root@asterisk ~]# yum install zlib
```

```
[root@asterisk ~]# yum install zlib-devel
```

```
[root@asterisk ~]# yum install openssl
```

```
[root@asterisk ~]# yum install openssl-devel
```

```
[root@asterisk ~]# yum install gnutls-devel
```

```
[root@asterisk ~]# yum install gcc
```

```
[root@asterisk ~]# yum install gcc-c++
```

- e. Desempaquetamos e instalamos los archivos en el siguiente orden**

**Instalamos Dahdi y Dahdi Tools:**

```
tar -xvf dahdi-linux-2.1.0.4.tar
```

```
tar -xvf dahdi-tools-2.1.0.2.tar
```

```
cd dahdi-linux-2.1.0.4
```

```
make clean
```

```
make
```

```
make install
```

```
cd dahdi-tools-2.1.0.2
```

```
./configure
```

```
make
```

```
make install
```

```
make config
```

### **Instalamos libpri:**

```
tar -xvf libpri1.4.9.tar
```

```
cd libpri1.4.9
```

```
make clean
```

```
make
```

```
make install
```

### **Instalamos asterisk:**

```
tar -xvf asterisk-1.6.0.9.tar
```

```
cd asterisk-1.6.0.9
```

```
make clean
```

```
./configure
```

```
make
```

```
make install
```

```
make samples
```

```
make config
```

### **Instalamos Asterisk addons**

```
tar -xvf asterisk-addons-1.6.1.0.tar
```

```
cd asterisk-addons-1.6.1.0
```

```
./configure
make
make install
```

### 2.3.5. Configuración de archivos principales de Asterisk

#### 2.3.5.1. Configuración del driver Dahdi

Hasta fines del año 2007 este driver era llamado zaptel y la configuración lógica de los canales análogos y digitales estaban en el archivo zapata.conf, la empresa Zaptel reclamó la marca como suya, motivo por el cual Digium, la empresa que revolucionó Asterisk, decide dejar de lado al antiguo driver zaptel y denominar al nuevo driver DAHDI (Digium Asterisk Hardware Device Interface).

**zaptel.conf** es sustituido por **/etc/dahdi/system.conf**

**zapata.conf** es sustituido por **/etc/asterisk/chan\_dahdi.conf**

#### 2.3.5.2. Herramientas del driver Dahdi para verificación de canales

dahdi\_cfg: Configura el archivo system.conf

dahdi\_genconf: Genera el archivo por defecto /etc/dahdi/system.conf

dahdi\_hardware: Muestra la lista de hardware detectado

dahdi\_monitor: Monitorea la señal de un canal análogo

dahdi\_scan: Genera lista de los canales DAHDI con algunos detalles

dahdi\_tool: Herramienta que describe las características de las tarjetas

#### 2.3.5.3. Descripción de parámetros principales del archivo /etc/dahdi/system.conf

##### 2.3.5.3.1. Configuración de las troncales E1 acceso primario

<Span Configuration>.

span=<span num>,<timing source>,<line build out>,<framing>,<coding>,<crc4>

<span num>: Numero secuencial del E1 configurado

<timing source>: Se escoge 1 para que la central de la RTPC sea la fuente maestra

<line build out (LBO)>: Se escoge el valor 0 a menos que sobrepase la distancia de 40 m hacia el modem conversor de E1 a Ethernet.

<framing>: Para un acceso primario se escoge ccs (señalización por canal común)

<coding>: Para un acceso primario se escoge hdb3

<crc4>: Indicar si se desea corrección de errores. normalmente esta deshabilitado

bchan: Canales de voz  
 dchan: Canal de señalización

Ejemplo de la configuración de un primario:

span=1,0,0,ccs,hdb3,crc4

bchan=1-15

dchan=16

bchan=17-31

#### 2.3.5.3.2. Configuración de los puertos FXS y FXO

Si además se tiene puertos FXO y FXS se verá reflejado lo siguiente:

fxoks=32-33

fxsks=34-35

#### 2.3.5.3.3. Carga de tonos

Cargamos los tonos de acuerdo a la zona de trabajo

defaulttone=es

loadzone=es

#### 2.3.5.3.4. Cancelación de eco

Si es que la tarjeta de puertos E I no incluye la cancelación de eco, podemos aplicar cancelador de eco por software.

echocanceller=<echocanceller name>,<channel(s)>

<echocanceller name>: Se tiene los siguientes mg2, kb1, sec2, y sec

<channel(s)>: Se describe los canales a los cuales se le quiere aplicar la cancelación de eco. Ejemplo:

echocanceller=mg2,1-15,17-31

#### 2.3.5.4. Configuración de diferentes tipos de canales

En los sistemas basados en Asterisk un canal es un medio de comunicación entre recursos internos /o externos, tales como líneas telefónicas análogas o digitales, un teléfono IP o una extensión física del sistema. Cada llamada controlada por el sistema viaja por su propio canal. Por consiguiente cada canal es nombrado por la interface o protocolo en el que se soporta el número del mismo. Así por ejemplo un canal SIP se podría

denominar SIP/123, en el caso de un canal conectado a una interfaz Dahdi se denominaría Dahdi/X donde X es el número físico del canal sobre la interfaz Dahdi correspondiente.

Una de las mayores habilidades del sistema Asterisk es la de traducir la información de un tipo de canal a otro de forma transparente, permitiendo la comunicación por ejemplo entre FXS a SIP, FXO a SIP, SIP a IAX, SIP a Dahdi, etc.

Los diferentes tipos de canales soportados por Asterisk se configuran en sus archivos de configuración respectivo, tenemos por ejemplo los siguientes:

h323.conf, sip.conf, chan\_dahdi.conf, iax.conf, mgcp.conf, etc.

#### **2.3.5.4.1. Descripción de parámetros principales del archivo chan\_dahdi.conf**

Aquí mostramos los parámetros principales del archivo chan\_dahdi.conf y la configuración ejemplar de un sistema Asterisk con una tarjeta interfaz primario y una tarjeta con dos interfaces FXS y dos FXO.

```
//Configuración global de los canales:
```

```
[channels]
```

```
language=en
```

```
context=incoming
```

```
usecallerid=yes
```

```
hidecallerid=no
```

```
callwaiting=yes
```

```
callwaitingcallerid=yes
```

```
threewaycalling=yes
```

```
transfer=yes
```

```
canpark=yes
```

```
echocancel=yes
```

```
faxdetect=both
```

```
callprogress=yes
```

```
inmediate=no
```

```
//Configuración de un primario:
```

```
context=incoming_pstn
```

```
switchtype=euroisdn
```

```

signalling=pri_cpe
group=1
channel=> 1-15.17-31

```

//Configuración de los puertos FXO y FXS:

```

context=incoming_fxo
signalling=fxs_ks
answeronpolarityswitch=yes
group=1
channel=> 32-33
context=incoming_fxs
signalling=fxo_ks
answeronpolarityswitch=yes
faxdetect=both
group=1
channel=> 34-35

```

#### 2.3.5.4.2 Configuración de canales SIP

Los canales SIP (Session Initial Protocol) son los canales utilizados por los dispositivos IP que utilizan este protocolo. Su configuración se encuentra en el archivo sip.conf. A continuación se muestra las opciones generales para configurar este archivo.

##### a.- Opciones generales:

**allow:** Permite el uso de codecs en orden de preferencia, dejar por defecto en all

**binaddr:** Es la dirección IP desde donde se permiten las conexiones SIP, dejar en 0.0.0.0 por defecto para no filtrar dirección IP alguna

**bindport:** El puerto en el que se escuchan las conexiones SIP es por defecto 5060

**language:** Define el idioma de los archivos de audio sobre canales SIP, dejar en es (español)

**relaxdtmf:** Funciona cuando Asterisk no reconoce los tonos DTMF dejar en yes

**videosupport:** Dejar activo cuando el dispositivo IP en SIP soporta video

##### b.- Opciones de dispositivos:

**callerid:** Identificación del llamante

**callgroup:** Grupo de llamadas al que pertenece el dispositivo SIP

**pickupgroup:** Grupo de captura de llamadas

**context:** Contexto asignado al dispositivo

**defaultip:** Asigna una dirección IP al dispositivo

**dtmfmode:** Define el modo de envío de tonos DTMF, dejar por defecto RFC2833

**language:** Define lenguaje de audio de dispositivo, dejar en es (español)

**secret:** Define la contraseña para que este dispositivo se registre. Es necesario que sea igual a la contraseña que se configura en el dispositivo SIP (teléfono IP)

**type:** Define el tipo de dispositivo, existen tres posibilidades: peer (recibe llamadas), user (hace llamadas) y friend (recibe y hace llamadas)

**username:** Nombre de usuario con el cual se registra el dispositivo SIP, va de la mano con la contraseña.

Asimismo existen diferentes tipos de canales que no serán materia de estudio en este informe, ya que en este caso solo se necesita de una tarjeta de 4 puertos FXS y 4 FXO para las extensiones análogas (fax y base móvil para llamadas a móviles) y una tarjeta con una interfaz de acceso primario a la RTPC, todos estos puertos son canales Dahdi. Los anexos internos serán teléfonos IP los cuales usarán canales SIP.

### 2.3.6. Plan de Marcación

El plan de marcación es el encargado de enrutar cada llamada en el sistema desde la fuente de la misma a través de diferentes aplicaciones a su destino final. La configuración del plan de marcación del sistema Asterisk se realiza en el siguiente archivo: `extensions.conf`.

El plan de marcación es la pieza central en cualquier sistema de telefonía, este especifica la manera en que se maneja el flujo de llamadas entrantes y salientes de llamadas, consiste en una lista de pasos o instrucciones que el sistema debe seguir.

#### 2.3.6.1. Fundamentos del Plan de Marcación

La mayor parte del plan de marcación se define en el archivo de configuración `extensions.conf`, este archivo está compuesto de cuatro componentes principales: contextos, extensiones, prioridades y aplicaciones. A continuación se detallan cada uno de estos conceptos.

### 2.3.6.1.1. Contextos

Los contextos definen la organización y alcance de la información o parámetros del plan de marcación definidos en cada uno de ellos. Se puede imaginar a los contextos como habitaciones selladas que tienen un comportamiento independiente.

En la práctica los contextos pueden ser utilizados para hacer que un sistema Asterisk conteste de forma diferente cada línea telefónica. Todas las llamadas que ingresan a un sistema Asterisk, se inician en un contexto y las instrucciones definidas en ese contexto determinarán lo que sucederá con esa llamada. Los contextos se definen por su nombre entre corchetes, todas las instrucciones colocadas después de la definición del contexto se considerarán parte del contexto hasta que se defina uno nuevo.

A fin de crear los esquemas de comunicación entre contextos se crean las inclusiones, se pueden incluir tantos contextos como se requiera creando así la comunicación entre ellos. Ejemplo:

```
[entrante]
include=> from_pstn
include=> from_fxo
```

En este ejemplo incluimos los contextos `from_pstn` y `from_fxo` en el contexto `entrante`

### 2.3.6.1.2. Extensiones

Dentro de cada contexto se definen una o más extensiones, estas determinan el flujo de la llamada al ser un conjunto de acciones. A simple vista pueden definirse como las extensiones telefónicas de una central PBX, pero su uso va más allá en los sistemas Asterisk, pueden ser constantes, patrones o literales.

Existe la extensión especial 's' que representa el inicio o start de una llamada.

Ejemplo:

```
[entrante]
exten=>s, 1, Answer()
```

### 2.3.6.1.3. Prioridades

Las prioridades son pasos numerados para definir el orden de ejecución en cada extensión. Cada prioridad puede llamar a una aplicación específica. Típicamente estos números de prioridades empiezan en 1 e incrementan sucesivamente con cada línea del contexto, o simplemente utilizan la letra n para indicar la prioridad siguiente. Ejemplo:

[entrante]

exten=>1,1,Answer()

exten=>1,2,Hangup()

#### 2.3.6.1.4. Aplicaciones

Las aplicaciones son los programas que realizan ciertas acciones tales sobre un canal de voz, tales como reproducir un sonido, aceptar tonos, colgar una llamada, etc. A las aplicaciones se le pueden pasar argumentos que afecten la forma en que realizan sus acciones.

A continuación se expone la lista de aplicaciones más útiles en un sistema Asterisk.

**Answer().-** Esta aplicación se utiliza para contestar una llamada sobre un canal cuando éste está timbrando.

**Playback().-** Ofrece la capacidad de reproducir un archivo de sonido previamente grabado y almacenado en el sistema. Con esta aplicación se ignora cualquier tono marcado por el usuario. El directorio donde se guardan los archivos de sonidos es el siguiente: /var/lib/asterisk/sounds/

**Hangup().-** Cuelga un canal activo, se utiliza al final del contexto una vez que deseamos liberar la llamada del sistema.

**Background().-** Esta aplicación permite interactuar con el sistema desde una llamada telefónica, en medida que reproduce un sonido pero se detiene ante la marcación de un tono del usuario y direcciona la llamada a la extensión marcada por el llamador.

**GoTo().-** Permite saltar de contexto, extensión o prioridad al contexto, extensión o prioridad especificado como argumento

**Dial().-** Esta aplicación nos permite comunicarnos o establecer la llamada con el canal indicado como argumento.

**While().-** Permite crear bucles que se ejecutan mientras una condición sea verdadera.

**GoToIf().-** Permite analizar una expresión y con base en el resultado cambiar el flujo de una llamada

**NoOp().-** En la línea de comandos muestra el valor de una variable a la consola de Asterisk

**Record().-** Permite grabar un archivo de sonido de lo hablado por el auricular.

**VoiceMail().-** Permite dejar mensajes en un buzón de voz

**VoiceMailMain().-** Verifica los mensajes de voz

**Wait().-** Espera un tiempo dado en segundos

### 2.3.6.1.5. Patrones de Numeración

El plan de numeración no está limitado solo a números fijos, se pueden utilizar también patrones de números para controlar el flujo de llamadas. Para esto el número de extensión debe iniciar con el símbolo “\_”. El sistema reconoce los siguientes caracteres:

X : corresponde a cualquier dígito entre 0 y 9

Z : corresponde a cualquier dígito entre 1 y 9

N : corresponde a cualquier dígito entre 2 y 9

[1237-9] : corresponde a los dígitos 1, 2, 3, 7, 8 y 9

. : corresponde a uno o más caracteres

### 2.3.6.1.6. Variables

Una variable es un contenedor para almacenar un dato. Pueden contener números, letras, caracteres alfanuméricos y son útiles para crear reglas y cambiar el flujo de las llamadas en el plan de marcación. Las variables definidas por el usuario no requieren que sean escritas en mayúsculas, aunque se recomienda. Por el contrario las variables predefinidas por Asterisk como por ejemplo `EXTEN` siempre se escriben con mayúsculas. Se denotan de la siguiente forma, con el signo \$ y el nombre entre llaves {}. Hay variables globales y de canal. Las globales se definen en el contexto [global] y son válidas en todo el plan de marcación, las de canal mientras dure la llamada en ese canal.

## 2.4. Protocolos de Voz sobre IP

### 2.4.1. Arquitectura de protocolos de VoIP

En la figura 2.6 se muestra la estructura de los protocolos usados en VoIP, donde se puede diferenciar los protocolos de señalización, transporte y control.

### 2.4.2. Protocolos de señalización

De acuerdo a la UIT, el protocolo de señalización se encarga de los mensajes y procedimientos utilizados para establecer una comunicación, negociar la tasa de bits, obtener el estado de los extremos y desconectar la llamada.

#### 2.4.2.1. H323

H323 es un estándar que norma los procedimientos para lograr sistemas audiovisuales multimedia, por lo que engloba varios estándares. Uno de estos procedimientos es la señalización de la llamada. Propone dos tipos de señalización:

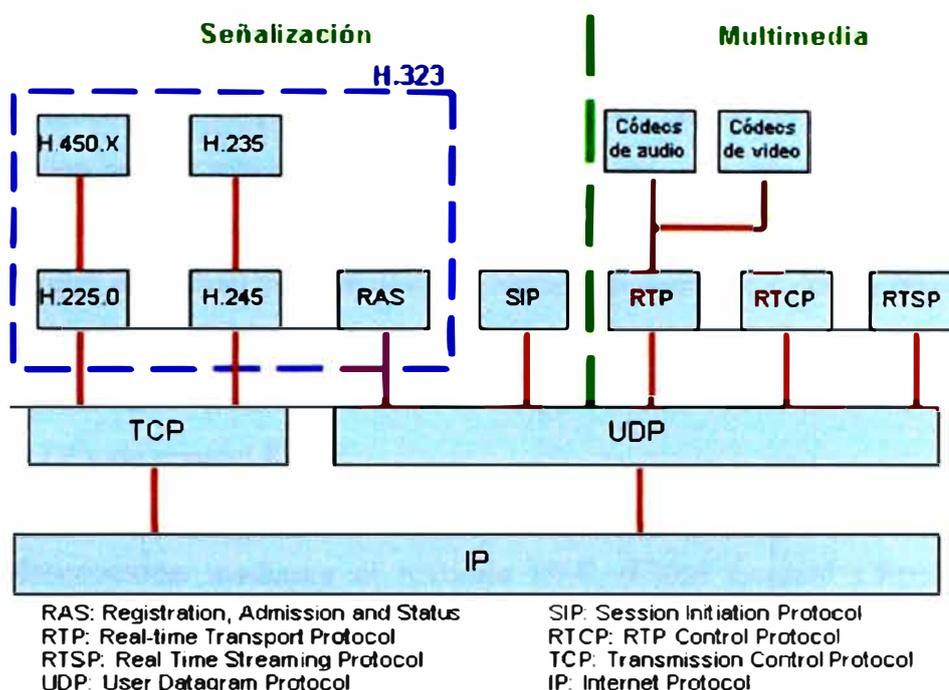


Figura 2.6. Esquema de protocolos VoIP

- **Señalización de control de llamada (H.225):** Este protocolo tiene dos funcionalidades. Si existe un gatekeeper en la red, define como un terminal se registra en él. Si no existe un gatekeeper define la forma como dos terminales pueden establecer o terminar las llamadas entre sí, en este último caso se basa en la recomendación Q931.
- **Señalización de control de canal (H.245):** Una vez que se ha establecido la conexión entre dos terminales usando H225, se usa el protocolo H245 para negociar y establecer los canales lógicos a través de los cuales se transmite la media. Define intercambio de capacidades (tasa de bits, codecs, etc.)

Se usa RAS siempre y cuando exista un gatekeeper en la red. El gatekeeper es un componente opcional cuya función principal es el control de admisión. A continuación se describe las fases de señalización de una llamada.

- **Establecimiento de la comunicación:** Primero se tiene que registrar y solicitar admisión al gatekeeper, para lo cual se usan los mensajes RAS. Luego el usuario llamante envía el mensaje SETUP, el llamado contesta con un Call

Proceeding. Para poder seguir con el proceso este usuario también tiene que registrarse y pedir admisión en el gatekeeper, una vez admitido envía el mensaje Alerting, cuando el llamado descuelga se envía el mensaje Connect.

- **Señalización de control:** En esta fase se abre la negociación mediante el protocolo H245 (control de canal). El intercambio de los mensajes establece quien será el maestro y quien el esclavo, así como también las capacidades y codecs de audio y video soportados (mensaje TCS, Terminal Capability Set) Como paso final se abre el canal lógico de comunicación (mensaje OLC Open Logical Channel).
- **Audio:** Los terminales envían la media a través de los protocolos de transporte RTP y de control RTCP.
- **Desconexión:** Cualquiera de los interlocutores puede iniciar el proceso de desconexión mediante el mensaje CLC (Close Logical Channel). Una vez hecho esto, ambos informan al gatekeeper de la finalización de la llamada a través de los mensajes RAS DCF (Disengage Confirm) y DRQ (Disengage Request).

La figura 2.7 muestra los pasos mencionados anteriormente.

#### 2.4.2.2. SIP (Session Initial Protocol)

SIP es un protocolo desarrollado por la IETF (Internet Engineering Task Force), basado en texto plano, contiene las normas para iniciar, modificar y finalizar una sesión de usuarios interactivos que implica elementos multimedia como video, voz, mensajería instantánea, juegos en línea, y la realidad virtual. Inicialmente se publicó en 1996 como RFC 2543, ahora obsoleto, debido a la publicación de la RFC 3261 en 2002.

El objetivo principal de SIP es la comunicación entre dispositivos multimedia. SIP hace posible la comunicación gracias a dos protocolos: RTP / RTCP y SDP.

RTP Protocolo se utiliza para el transporte de voz y datos en tiempo real (al igual que el protocolo H.323), mientras que el protocolo SDP se utiliza para negociar las capacidades de los participantes, el tipo de codificación, etc.

SDP (Session Description Protocol) es usado para describir sesiones multicast en tiempo real usado para invitaciones y anuncios de parámetros de una sesión multimedia.

SIP ha sido diseñado en conformidad con el modelo de Internet. Es decir es un protocolo cu\_ a señalización es orientado extremo a extremo lo que significa que toda la

lógica se almacena en los dispositivos finales (excepto el enrutamiento de mensajes de SIP).

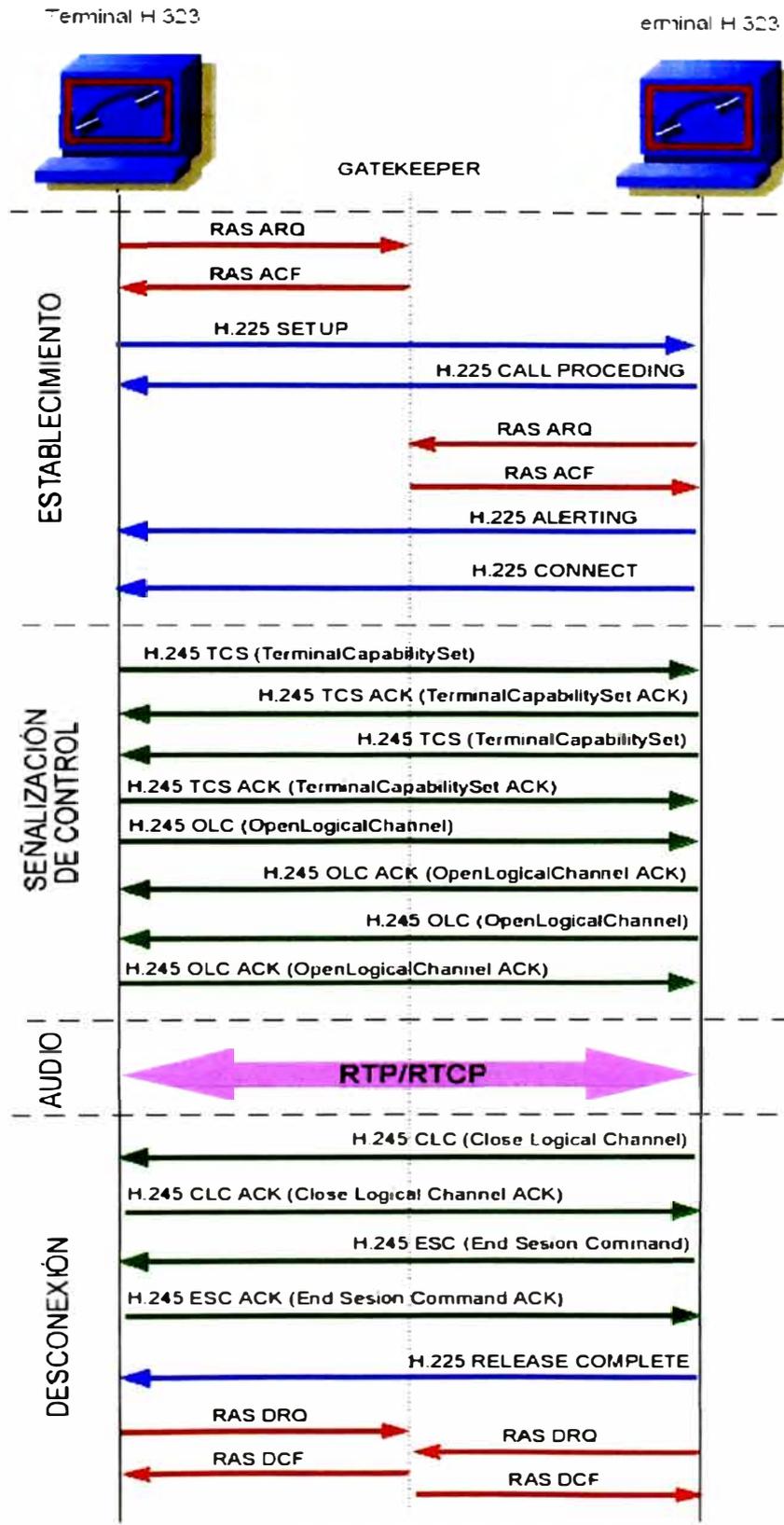


Figura 2. . Flujo de señalización de llamada en H323

Por lo tanto, SIP es un protocolo de control de la capa de aplicación, un protocolo de señalización para la telefonía de Internet. SIP puede establecer sesiones para intercambio de audio y videoconferencias, juegos interactivos, desvío de llamadas que se desplegarán sobre redes IP, lo que permite integrar a los proveedores de servicios básicos de telefonía IP con servicios Web, correo electrónico y chat. Se basa en mensajes de petición y respuesta y reutiliza muchos conceptos de las anteriores normas, como HTTP y SMTP. Los componentes presentes en SIP son los siguientes:

**a.- Agentes de Usuario (User Agent UA):** Existen dos tipos de agentes de usuario que siempre están presentes en las comunicaciones cliente servidor.

- Agente de usuario cliente (UAC), el agente cliente genera peticiones SIP y recibe respuestas.
- Agente de usuario servidor (UAS), El UAS responde las peticiones SIP

**b.- Servidores SIP:** Existen tres clases de servidores que son las siguientes:

- Servidor de redirección (Redirect Server), Reencamina las peticiones que recibe a su próximo servidor
- Servidor Proxy (Proxy Server), Corren un programa intermediario para actuar tanto de servidor como cliente y poder establecer las llamadas entre usuarios.
- Servidor de Registro (Register Server), Hace la correspondencia entre direcciones IP y usuarios SIP.

Se definen dos tipos de mensajes en SIP:

a.- Peticiones SIP: Se definen seis mensajes básicos.

- INVITE: Permite invitar a un usuario a iniciar una sesión, o modificar parámetros de una sesión ya establecida.
- ACK: Confirma el establecimiento de la sesión.
- OPTION: Solicita información de algún servidor en particular.
- BYE: Finaliza una sesión
- CANCEL: Cancela una petición pendiente
- REGISTER: Registra al agente usuario

b.- Respuestas SIP: Se definen seis tipos de respuesta, se diferencian por el primer dígito de su código.

- 1XX Mensajes provisionales
- 2XX Respuestas de éxito
- 3XX Respuestas de redirección
- 4XX Respuestas de falla de método
- 5XX Respuestas de falla de servidor
- 6XX Respuestas de fallas globales

En la figura 2.8 se muestra el establecimiento de una llamada entre dos usuarios SIP registrados en un servidor.

Las dos primeras transacciones tienen que ver con el registro de usuarios con el punto intermedio que es el servidor. La siguiente transacción establece el inicio de sesión, el usuario A envía un INVITE al usuario B que es redireccionado por el servidor. La sesión se establece cuando ambos puntos envían la confirmación. Mediante el protocolo SDP se describe los codecs, tipo de data (voz, video o datos) y otros parámetros de la comunicación; cuando la sesión se ha establecido, la media se envía vía el protocolo RTP. Cuando alguien quiere finalizar la comunicación envía el mensaje BYE dando luego la confirmación para el término de la sesión.

#### **2.4.2.3. Diferencias entre H323 y SIP**

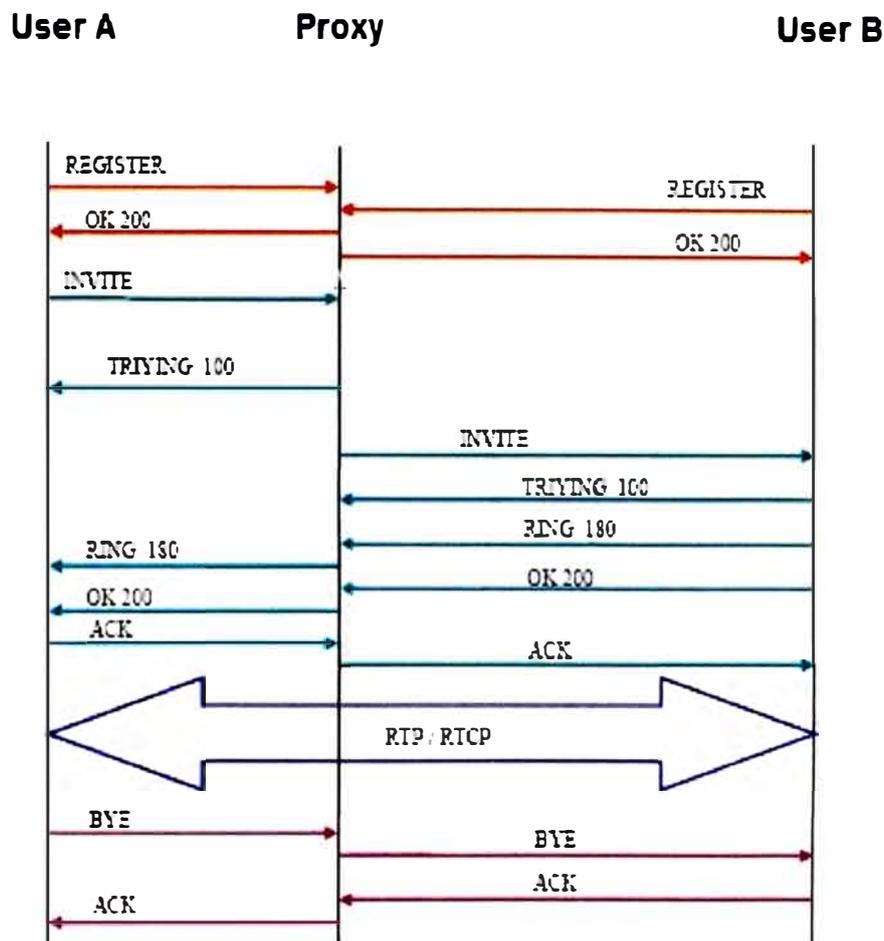
Entre las diferencias se puede mencionar la velocidad para establecer una comunicación, SIP utiliza UDP en cambio H323 utiliza TCP para establecer una comunicación con H225 y H245, por lo que se nota la rapidez de SIP.

Otra diferencia notable es que H323 define canales lógicos antes de enviar datos, en cambio SIP solo negocia codecs e información de la sesión pero no asegura un canal de comunicación ni separa una tasa de bits para la comunicación.

#### **2.4.2.4. IAX**

El protocolo IAX cuyo significado de sus siglas es Inter-Asterisk eXchange, nació con la necesidad de interconectar servidores Asterisk y de hacer un protocolo de VoIP que trabaje bien en redes con nat firewalls y que pueda integrar el audio y la señalización de varias llamadas a través de un solo stream, minimizando el consumo de ancho de banda en las comunicaciones de voz y multimedia de VoIP. Fue desarrollado por Mark Spencer, el

actual protocolo es el IAX2 que fue producto de los aportes de los colaboradores de la comunidad y que ha dejado obsoleto al antiguo protocolo IAX.



**Figura 2.8. Flujo de señalización de llamada en SIP**

IAX es un protocolo binario y no de texto como lo es el SIP, utiliza un solo puerto 4569 UDP, y trabaja bien en los ambientes NAT (el obsoleto protocolo IAX1 usaba el puerto 5036). IAX utiliza solamente un puerto de UDP para control y tráfico de datos, es decir con IAX siempre se tendrá audio si la conexión de control pudo ser establecida.

Entre las utilidades del IAX está el poder hacer la interconexión entre servidores Asterisk cuya denominación es “Trunking”, esta interconexión incluye autenticación; el trunking de IAX permite que los streams múltiples de voz compartan un solo “trunk” a otro servidor, reduciendo así las sobrecargas creadas por los paquetes IP, ya que solo se reserva el ancho de banda necesario en cada comunicación comparado con otros trunk TDMoIP que reservan un ancho de banda específico para cada canal. En la figura 2.9 se

muestra las fases y el intercambio de mensajes entre dos terminales IAX. Una llamada tiene 3 fases:

**a.- Establecimiento de la llamada:** El terminal llamado inicia la llamada con un mensaje NEW, al cual el llamado responde con un ACCEPT, a continuación el llamado envía un mensaje RINGING, el cual es confirmado por el llamante con un ACK. Por último el llamado acepta la llamada con un mensaje ANSWER y el llamante confirma con un ACK.

**b.- Flujo de datos o audio:** Se envían las tramas M y F. Las tramas M son mini-tramas que tienen una cabecera de solo 4 bytes para reducir el ancho de banda consumido, mientras que las tramas F son tramas completas que llevan información de sincronización. Es importante señalar que ambos flujos se envían a través del protocolo UDP puerto 4569 para evitar los problemas con el NAT.

**c.- Liberación de la llamada:** Se envía el mensaje HUNGUP y se confirma.

Las tramas que se envían en IAX son binarias y cada conjunto de bits tiene un significado, existen dos tipos de tramas, las tramas F o Full Frames y las M o Mini Frames. En la figura 2.9 se muestra una gráfica del intercambio de mensajes para el establecimiento de una llamada en IAX.

**a.- Tramas F o Full Frame:** Estas tramas deben ser respondidas explícitamente. En la figura 2.10 se muestra el formato de una trama F. El significado de cada uno de los campos es el siguiente:

- \* **F:** Un bit que indica si la trama es F (full frame) o no. Para que sea F o full frame debe estar puesta a 1.
- \* **Source Call Number:** Número de llamada de origen de 15 bits que identifican la conversación de origen ya que puede haber varias comunicaciones multiplexadas por la misma línea.
- \* **R:** Bit de retransmisión. Se pone a uno cuando la trama es retransmitida.
- \* **Destination Call Number:** Número de llamada destino.
- \* **Timestamp o sello de tiempo:** Para marcar el tiempo en cada paquete.
- \* **Oseqno:** Secuencia de salida. Número de secuencia de salida con 8 bits. Comienza en 0 y va incrementándose en cada mensaje.
- \* **ISeqno:** Secuencia de entrada.
- \* **Frame Type:** Indica la clase de trama.

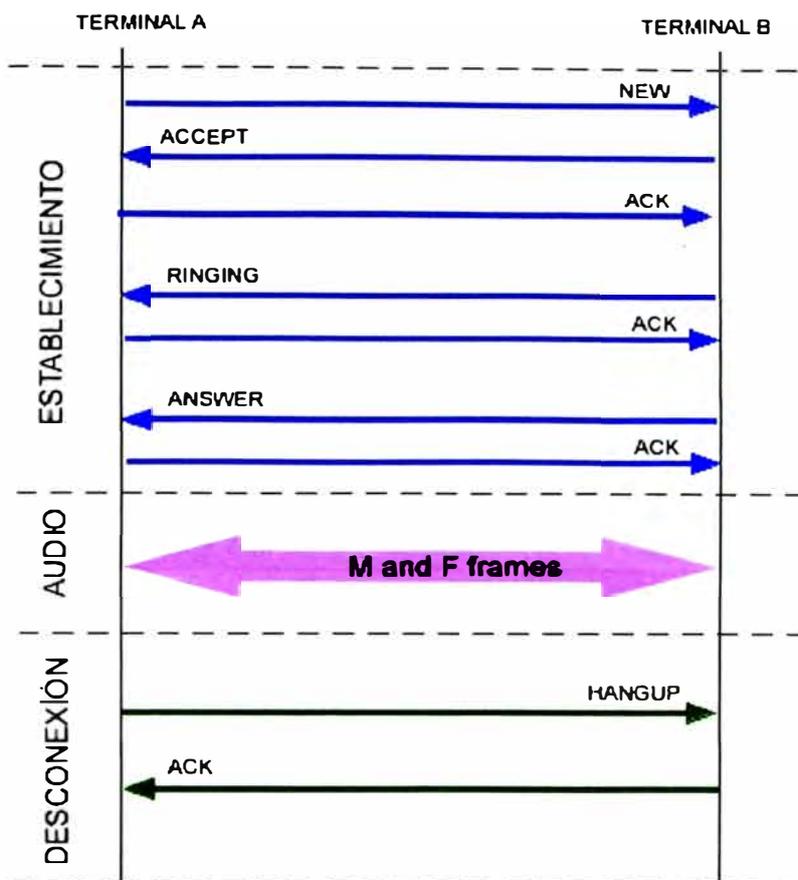


Figura 2.9. Intercambio de mensajes IAX

F	Source Call Number- Numero de llamada de origen	R	Destination Call Number- Numero de llamada destino
Timestamp- Sello de tiempo			
OSeqno- sec. de salida	ISeqno- sec. de entrada	Frame Type - Tipo de trama	C Subclass- subclase
Data- Datos			

Figura 2.10. Formato de una trama F

- \* C: Puesto a 0 indica que el campo subclase debe tomarse como 7 bits (un solo mensaje); Puesto a 1 indica que el campo subclase se obtiene con 14 bits (dos mensajes consecutivos).

- \* **Subclass:** Subclase del mensaje.
- \* **Data:** Datos que se envían en formato binario.

**b.- Trama M o Mini Frame:** Estas tramas se envían sin necesidad de ser respondidas y con la menor cabecera posible, si se pierde alguna no causa mayor impacto. Los campos son parecidos al de las tramas F, el bit F está puesto a 0 y el timestamp está truncado a 16 bits para aligerar la cabecera. El campo Type Frame y el Subclase determinan de que tipo y que información contiene una trama F. En las tablas 2.3, 2.4, 2.5 y 2.6 se describen los diferentes valores y tipos de trama F que sirven como control y señalización.

#### **2.4.2.5. Diferencias entre IAX y SIP**

Entre las diferencias podemos mencionar que IAX utiliza un menor ancho de banda que SIP ya que IAX es un protocolo binario, en cambio SIP es de texto, además IAX intenta reducir al máximo la información de las la cabeceras de tramas IP. Con respecto al nat, IAX trabaja muy bien en redes NAT ya que toda la información (voz y señalización) es transmitida por un único puerto, en cambio SIP envía estos flujos de forma distinta pudiendo tener problemas para superar los router y firewall. SIP es un protocolo estandarizado por la IETF hace bastante tiempo y es ampliamente implementado por todos los fabricantes de equipos y software. IAX está aun siendo estandarizado y es por ello que no se encuentra en muchos dispositivos existentes en el mercado. IAX utiliza un solo puerto (4569) para mandar la información de señalización y los datos de todas sus llamadas. Para ello utiliza un mecanismo de multiplexión o "trunking". SIP, sin embargo utiliza un puerto (5060) para señalización y 2 puertos RTP por cada conexión de audio.

#### **2.4.3. Protocolos de transporte**

##### **2.4.3.1. RTP (Real -Time Transport Protocol)**

Este protocolo define un formato de paquete para llevar audio y video a través de Internet. Está descrito en la RFC3550. No usa un protocolo UDP determinado, la única regla que sigue es que las comunicaciones UDP se hacen vía un puerto impar y el puerto par consecutivo se usa para el control RTP (RTCP). Las aplicaciones que usan RTP son menos sensibles a la pérdida de paquetes, pero si lo son a los retardos por lo que se usa UDP para estas aplicaciones. Por otro lado RTP no brinda proporción de servicio, pero se puede resolver a través de otros mecanismo como el marcado de paquetes en los nodos de la red.

Tabla 2.3. Valores para el campo Type Frame de una trama F

Type Frame	Descripción	Detalles
00000001	DTMF	Sirve para enviar dígitos DTMF
00000002	Datos de voz	El campo subclase indica el tipo de codec de audio que se utiliza
00000003	Datos de video	El campo subclase indica el tipo de codec de video que se utiliza
00000004	Control	Mensajes de control de sesión. Sirve para controlar el estado de los dispositivos finales.
00000005	No usado	
00000006	Control IAX	Mensajes de control del protocolo IAX. Gestiona las interacciones necesarias entre los dispositivos finales.
00000007	Texto	
00000008	Imagen	
00000009	HTML	

Tabla 2.4. Codec utilizado de acuerdo a subclase para Type Frame 0x02

Valor subclase (Type Frame =0x02)	Codec que se utiliza en la conversación
0x0001	G.723.1
0x0002	GSM
0x0004	G.711 u (u-law)
0x0008	G.711 a (a-law)
0x0080	LPC10
0x0100	G.729
0x0200	Speex
0x0400	iLBC

Tabla 2.5. Mensajes de señalización de acuerdo a subclase para Type Frame 0x04

Valor subclase (Type Frame =0x04)	Descripción	Detalles
0x01	Hangup	La llamada se ha colgado
0x02	Ring	El teléfono esta sonando
0x03	Ringin (ringback)	
0x04	Answer	Respuesta
0x05	Busy Condition	El usuario está ocupado
0x08	Congestion	Existe congestión
0x0e	Call Progress	Progreso de la llamada

Tabla 2.6. Significado de valores de subclase para Type Frame 0x06

Valor subclase (Type Frame =0x05)	Descripción	Detalles
0x01	NEW	Iniciar una nueva llamada
0x02	PING	Enviar un ping
0x03	PONG	Responder un ping
0x04	ACK	Respuesta afirmativa
0x05	HANGUP	Inicio de desconexión
0x06	REJECT	Rechazo
0x07	ACCEPT	Aceptación
0x08	AUTHREQ	Petición de autenticación
0x09	AUTHREP	Respuesta de autenticación
0x0a	INVAL	Llamada no válida
0x0b	LAGRQ	Petición de Lag
0x0c	LAGRP	Respuesta de Lag
0x0d	REGREQ	Petición de registro
0x0e	REGAUTH	Autenticación de registro
0x0f	REGACK	ACK de registro
0x10	REGREJ	Denegación de registro
0x11	REGREL	Liberación de registro
0x12	VNAK	Petición de retransmisión
0x13	DPREQ	Petición de dialplan
0x14	DPREP	Respuesta de dialplan
0x15	DIAL	Marcado
0x16	TXREQ	Petición de transferencia
0x17	TXCNT	Conexión de transferencia
0x18	TXACC	Aceptación de transferencia
0x19	TXREADY	Transferencia preparada
0x1a	TXREL	Liberación de transferencia
0x1b	TXREJ	Rechazo de transferencia
0x1c	QUELCH	Parar transmisión de audio
0x1d	UNQUELCH	Continuar transmisión de audio
0x20	MWI	Indicador de mensaje en espera
0x21	UNSUPPORT	Mensaje no soportado

#### 2.4.3.2. RTCP (Real -Time Transport Control Protocol)

Se basa en la transmisión de paquetes de control fuera de la banda a todos los nodos participantes en la sesión. Tiene tres funciones principales:

- Proporciona realimentación en la calidad de la data: indicadores de jitter, retardo

- Utiliza nombres canónicos (CNAME) para identificar a cada usuario en cada sesión
- Como cada usuario envía sus tramas de control a los demás, cada usuario sabe el número total de participantes, este número se usa para calcular la tasa a la cual se van a enviar los paquetes.

#### 2.4.4. CODEC

Significa Codificador-Decodificador, describe una implementación por software o hardware que permite una correcta transmisión de un flujo de datos. A continuación se describen algunos de los codecs de voz más conocidos

##### 2.4.4.1. G711 (PCMA o PCMu)

Tiene una tasa de transmisión alta (64 kbps), desarrollado por la UIT, es el codec nativo de redes digitales modernas de telefonía.

Tiene una tasa de muestreo de 8000 muestras por segundo, lo que permite un ancho de banda total para voz de 4000 Hz, cada muestra se codifica en 8 bits lo que lleva a una tasa de 64 kbps.

Existen dos versiones de este codec Ley-A y Ley-u, la segunda de ellas se usa en USA y en Japón. La diferencia entre ellas es la forma del muestreo de la señal. En las figuras 2.11, 2.12 y 2.13 se muestran las fórmulas de cada una de ellas.

Los valores de  $u$  y  $A$  están estandarizados por la UIT y son de  $u = 255$  y  $A = 100$ . La forma logarítmica refuerza las muestras más pequeñas con la finalidad de protegerlas del ruido.

Ley-A:

$$\begin{aligned} \circ \quad y &= \frac{Ax}{1 + \ln A} && \text{para } x \leq \frac{1}{A} \\ \circ \quad y &= \frac{1 + \ln Ax}{1 + \ln A} && \text{para } \frac{1}{A} \leq x \leq 1 \end{aligned}$$

Figura 2.11. Fórmula de la ley A

Ley- $\mu$ :

$$y = \frac{\ln(1 + \mu x)}{\ln(1 + \mu)}$$

Figura 2.12. Formula de la ley u

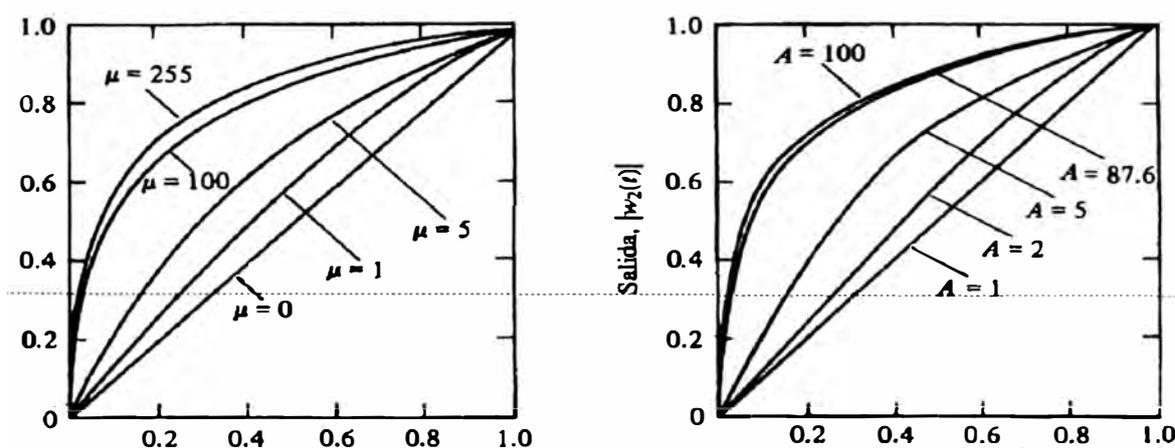


Figura 2.13. Gráfica comparativa entre Ley A y Ley u

El uso de G711 en VoIP ofrece la mejor calidad (no realiza compresión de voz). Esto se comprueba con la medida del MOS (Mean Opinión Score) que es una medida cualitativa de la voz. Un MOS de 5 indica la mejor calidad y 0 la peor. G711 tiene el MOS más alto de todos los codecs con un MOS de 4.1. También hay un menor retardo debido a que no hace uso extensivo del CPU para la compresión. El inconveniente principal es que necesita una mayor tasa de datos 80 kbps aproximadamente incluyendo la cabecera TCP/IP, pero esto no debería ser problema ante un ancho de banda grande.

#### 2.4.4.2. G729

Este codec comprime la señal en periodos de 10 mseg y realiza una compresión a una tasa de 8 kbps, se usa en las aplicaciones VoIP por su tasa pequeña de ancho de banda, idealmente tiene un MOS de 3.8. El uso de este codec requiere de una licencia.

Normalmente los tonos DTMF, los datos y fax no son transportados confiablemente por este codec, para esto se usa el G711. Tiene algunas variantes como por ejemplo el G729A,

que requiere menor actividad de procesamiento pero afecta en la calidad de la voz, el G729B que incluye supresión de silencios.

#### **2.4.4.3. GSM**

Este codec se llama oficialmente RPE-LTP (Regular Pulse Excitation - Long Term Prediction) pero se conoce mundialmente como GSM debido a que es el codec usado en el estándar GSM de comunicaciones móviles. Tiene una tasa de bits de 13 kbps y un MOS ideal de 3.6 y realiza la codificación generando coeficientes representativos en un tiempo determinado, este intervalo normalmente es de 20 mseg. En la tabla 2.7 se muestra una comparación entre los distintos codecs de voz.

#### **2.4.5. Hardware de telefonía IP**

El hardware mencionado son los equipos que utiliza el usuario para conectarse a la red IP, básicamente son dos, los adaptadores analógicos (ATA) y los teléfonos IP.

##### **2.4.5.1. Adaptadores Telefónicos Análogos**

Son dispositivos con una interfaz para conectar un teléfono analógico (conector RJ11) y otra interfaz para conectar a la red IP (conector RJ45). Existen marcas como Linksys, Grandstream, Quantum, DLINK, etc. Básicamente su función es la de proveer señalización FXO a los teléfonos, es decir se comporta como dispositivo FXS. A continuación se explica brevemente estos términos.

- FXO (Foreign Exchange Office), es la interfaz que se conecta a la red de telefonía RTPC o a una PBX, está presente en los teléfonos analógicos, recibe la señalización FXS.
- FXS (Foreign Exchange Station), es la interfaz que se conecta a un teléfono analógico y le da las señales de timbrado, voltaje, etc. En un escenario convencional de telefonía analógica, el FXS está en la central de conmutación, brindando señalización al dispositivo FXO (teléfono analógico)

Se tienen dos posibilidades para conectar teléfonos analógicos en una red IP, una es que la central IP tenga tarjeta con puertos FXS y la otra es usar gateways como los ATA (Analogue Telephone Adaptor) que sirvan como interfaz entre la señal análoga y los datos IP.

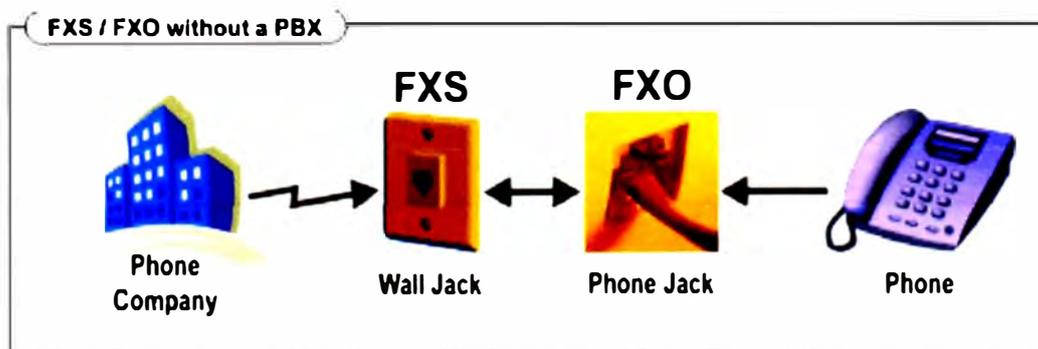
En los gráficos 2.14 y 2.15 se muestra diagramas de conexiones para entender que es un puerto FXS y que un puerto FXO.

**Tabla 2.7. Esquema comparativo entre diferentes tipos de codec**

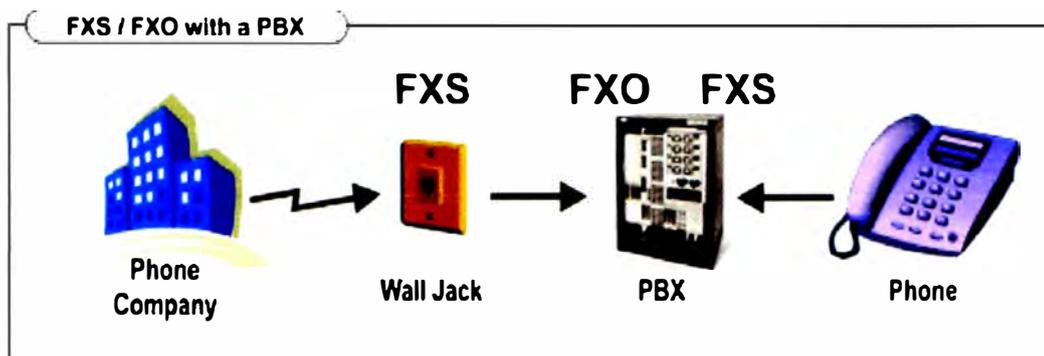
CODEC	DESCRIPCION	BIT RATE (kbps)	FRECUENCIA MUESTREO (KHz)	TAMAÑO DE CUADRO	OBSERVACION	MOS IDEAL
G.711	Pulse Code Modulation	64	8	Muestreada	Ley A - Ley u	4.1
G.729	Conjugate-structure algebraic-code-excited linear-prediction (CS-ACELP)	8	8	10	Bajo retardo 15 mscg	3.8
GSM	Regular Pulse Excitation - Long term Prediction (RPE-LTP)	13	8	20	Usado por GSM	3.5 - 3.7

#### 2.4.5.2. Teléfonos IP

Son dispositivos que soportan uno o varios protocolos de señalización. Entre las marcas mas conocidas están Cisco, Polycom, Snom, Grandstream, Atcom, etc. La gran mayoría soporta como mínimo el codec G711, pudiendo soportar otros con licencia. Integran también servicios suplementarios y existe una gran variedad diferenciándose en el precio por la cantidad y calidad de prestaciones que realiza.



**Figura 2.14. Puertos FXS y FXO, línea telefónica sin PBX**



**Figura 2.15. Puertos FXS y FXO, línea telefónica con PBX**

En la figura 2.16 se observa un teléfono IP de la marca Grandstream el cual es de buena calidad, costo económico y ofrece diferentes prestaciones.



**Figura 2.16. Teléfono IP, marca Grandstream, modelo BT200**

## CAPITULO III

### PLANTEAMIENTO DE SOLUCIÓN PARA UNA INSTALACIÓN DE ASTERISK CON 50 ABONADOS SIP Y ACCESO PRIMARIO

#### 3.1. Arquitectura de red

A continuación se muestra la solución planteada para esta empresa con 50 puestos de trabajo repartidos entre las áreas de Contabilidad, Ventas, Gerencia, Operaciones y Sistemas, utilizando la red Ethernet 10/100 Mbps ya implementada que consta de 3 switches de 24 puertos, un router para la salida a Internet, todas las PC's con su respectiva tarjeta de red Ethernet incorporada, un acceso primario alquilado a una empresa proveedor de telefonía digital, una línea analógica para fax, una base celular con dos puertos FXS para las llamadas a móviles.

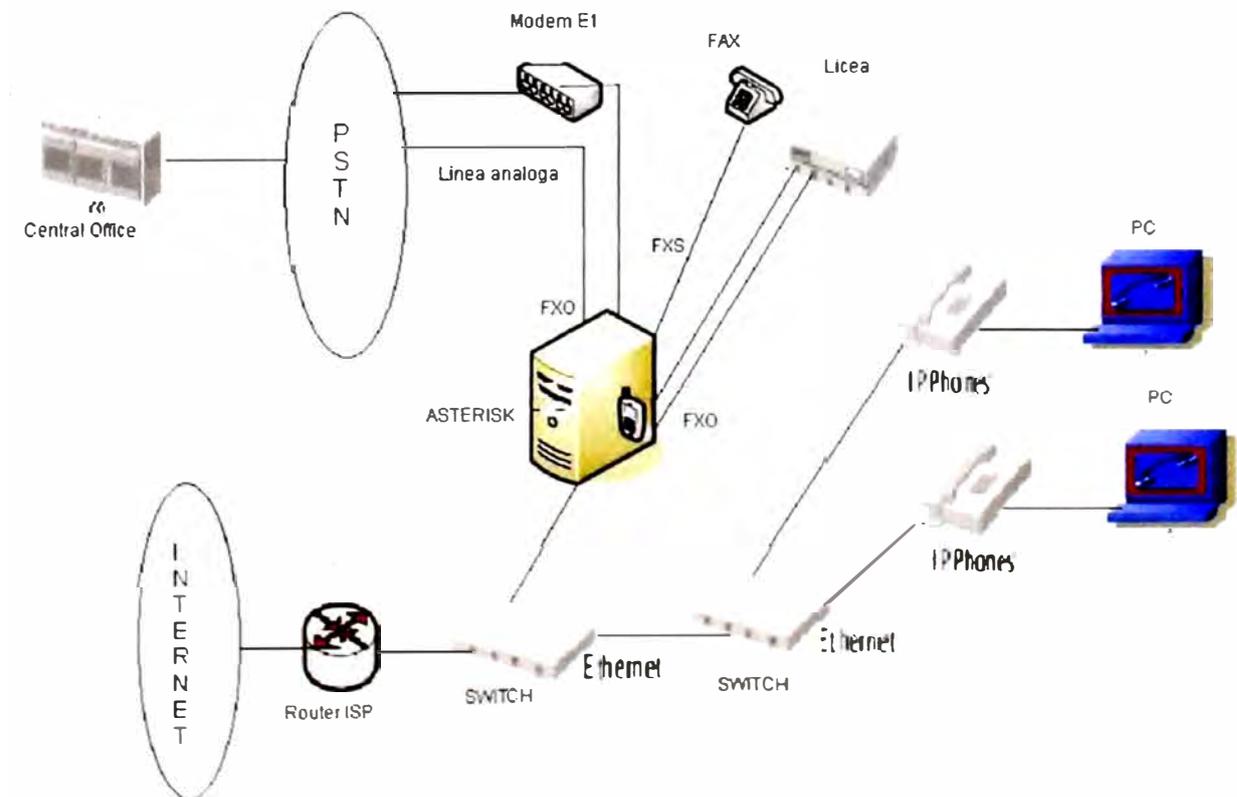


Figura 3.1. Arquitectura de la solución TDM + IP

## 3.2. Descripción de Hardware

### 3.2.1. Servidor

De acuerdo a los requerimientos y estimación del tráfico telefónico, se llegó a la conclusión que el máximo número de llamadas externas (hacia y desde la RTPC) simultáneas en horas pico serían 20 (20 Erlang) y de 8 llamadas internas, dando un total de 28 conexiones simultáneas en total. Para esta capacidad basta con un Pentium IV Dual Core que puede manejar hasta 60 llamadas simultáneas por experiencia. Ya se ha mencionado anteriormente que no existe alguna fórmula o regla en especial para elegir el hardware del servidor. Así elegimos un servidor con las siguientes características.

- Pentium IV Intel Dual Core CPU 2.4 Ghz
- 2 Discos SATA 200 Gb
- Memoria RAM DDR2 - 1 Gb
- Case con fuente redundante 550 Watts
- Video y Red incorporado
- 4 puertos USB, serial, 2 PS/2
- 4 ranuras para tarjetas PCI
- 1 ranura para tarjeta PCI Express

### 3.2.2. Tarjeta Digium 4 FXO + 4 FXS

Esta tarjeta es necesaria para conectar al Fax (1 FXS y 1 FXO) y a las liceas (2 FXO).

- TDM844BF de Digium con 4 puertos FXS y 4 puertos FXO, PCI de 3.3 voltios

### 3.2.3. Tarjeta Digium 1 E1

Esta tarjeta es necesaria para conectar a la RTPC a través de un acceso primario

- TE121PF de Digium con 1 span digital E1/T1/J1 PCI Express 3.3 voltios

### 3.2.4. Base celular

También llamada "licea" es para enrutar las llamadas a móviles, soporta dos chips y 2 puertos FXS

- ITS CGW-T de Audiocodes

### 3.2.5. Teléfonos IP

Teléfonos que soporten el protocolo SIP, de buena calidad y de costo no tan alto

- Teléfonos marca Grandstream modelo BT200, full dúplex, dos interfaces RJ45 10/100 Mbps

### 3.3. Comparación y evaluación de costos

A continuación se presenta el costo total para la implementación de acuerdo a cotizaciones actualizadas a la fecha del mes de mayo del 2009 con precios promedio de los componentes.

**Tabla 3.1. Cuadro de costos para la solución TDM + IP con Asterisk**

ITEM	CANTIDAD	P. UNITARIO	P. TOTAL
Servidor Pentium IV Dual Core 2.4 Ghz	1	1050	1050
Tarjeta Digium TDM844BF	1	690	690
TE121PF Digium con 1 span digital E1	1	550	550
Base celular ITS CGW-T 2 puertos FXS	1	400	400
Teléfonos IP Grandstream BT200	50	60	3,000
		<b>Total \$</b>	<b>5,690</b>

En el caso que la empresa hubiese adquirido una central PBX con similares capacidades de la marca propietaria NORTEL, obtendríamos los siguientes costos.

**Tabla 3.2. Cuadro de costos para la solución con PBX propietaria**

ITEM	CANTIDAD	P. UNITARIO	P. TOTAL
Central telefónica BCM400 Base System	1	4,628	4,628
Modulo Digital 01 PRI	1	1,380	690
Modulo 04 troncales analógicas	1	554	550
Modulo de anexos digitales	1	922	400
Teléfonos Digitales T7100	50	107	5,350
		<b>Total \$</b>	<b>11,618</b>

De la comparación se observa que adquirir una central PBX propietaria costaría casi el doble que al hacerlo con software libre y sin restricción alguna. Si siquiera aumentando un curso de capacitación para el personal de sistemas de la empresa, se llegarían a igualar lo costo

### 3.4. Cronograma de tiempos de ejecución

El tiempo de implementación estimado que toma desde el diseño hasta la puesta en servicio incluyendo pruebas y capacitación del personal de sistemas de la PYME es aproximadamente tres meses. Generalmente la etapa que toma mayor tiempo es la adquisición de equipos ya que la mayoría de ellos son importados. A continuación se muestra una tabla con el estimado de tiempos de ejecución.

**Tabla 3.3. Cuadro de tiempos para implementar el sistema de telefonía IP más acceso primario**

ACTIVIDADES	TIEMPO EN MESES		
	1	2	3
Diseño del Sistema	[Barra roja que cubre los meses 1 y 2]		
Adquisición de Equipos	[Barra azul que cubre los meses 1, 2 y 3]		
Instalación	[Barra verde que cubre los meses 2 y 3]		
Pruebas	[Barra morada que cubre los meses 2, 3 y 4]		
Capacitación	[Barra rosa que cubre los meses 3 y 4]		

### 3.5. Direccionamiento IP

Se puede escoger sin problemas una red clase C del rango privado 172.16.140.0 / 24 ya que con esto tenemos capacidad para 253 Host y lo requerido son 50 direcciones IP para teléfonos IP mas 50 direcciones IP para las PC's en total 100.

Para el caso de una pequeña empresa el tema de seguridad y calidad de servicio con VLAN se puede obviar, debido a que con un número de estaciones menor a 100 la performance de la red es buena y el tráfico cursado es menor a los 10/100 Mbps que ofrece una red Ethernet, no es necesario segmentar dominios de broadcast ni priorizar tráfico de voz. Cada llamada de VoIP con el codec G711 requiere una tasa de 88 kbps incluyendo la cabecera TCP/IP, de la estimación de llamadas simultáneas el total resultado 26 lo cual significaría un consumo de 2.28 Mbps de tráfico VoIP.

Entonces para mantener un orden, escogemos que las PC's de trabajo y servidores de aplicaciones que no sean voz, tengan direcciones IP del rango 172.16.140.2 – 172.16.140.101

Para los teléfonos IP escogemos el rango entre 172.16.140.102 – 172.16.140.254. Para el servidor Asterisk escogemos la IP 172.16.140.100

A continuación en la tabla 3.4 se muestra la asignación de IP por área de una forma organizada.

Tabla 3.4. Direccionamiento IP para PC's y teléfonos IP

Área / Posición / Servicio	IP de voz	IP de datos
Gerencia	172.16.140.2 - 172.16.140.10	172.16.140.102 - 172.16.140.110
Ventas	172.16.140.11 - 172.16.140.30	172.16.140.111 - 172.16.140.130
Contabilidad	172.16.140.31 - 172.16.140.50	172.16.140.131 - 172.16.140.150
Sistemas	172.16.140.51 - 172.16.140.70	172.16.140.151 - 172.16.140.170
Operaciones y Mantenimiento	172.16.140.71 - 172.16.140.90	172.16.140.171 - 172.16.140.190
Servidor Asterisk	172.16.140.100	172.16.140.100

### 3.6. Implementación de redundancia en el servidor

El servidor Asterisk es el núcleo de las comunicaciones de voz de la empresa, por lo tanto requiere tomar medidas con respecto a la alta disponibilidad que debe tener al menos uno de sus componentes. De todos los componentes, el elegido es el disco duro, ya que es el más proclive a sufrir daños indeseados por diferentes motivos y almacena toda la información relevante del sistema. Como recurso se implementará RAID (Redundant Array of Independent Disks) por software, que es un sistema de almacenamiento que usa múltiples discos duros en los cuales distribuye o replica la información. Existen distintos niveles RAID que se pueden implementar en el servidor y cada nivel es elegido de acuerdo a lo deseado en el sistema de almacenamiento. Para este sistema de telefonía se utilizará el nivel 1 de RAID, ya que este nivel mejora tanto el rendimiento (el sistema puede leer dos datos distintos en dos discos diferentes simultáneamente) y la confiabilidad de la información ya que la copia en cada uno de los discos (disco espejo); la probabilidad de falla de los discos a la vez es mucho menor que la probabilidad de falla de uno sólo de ellos.

Las distribuciones Linux con versión de núcleo superior a 2.4 cuentan con RAID por software, a continuación se muestra los pasos para configurar el RAID nivel 1 en los dos discos duros sda1 y sdb1 del servidor.

- Descargar el paquete raidtools e instalarlo.

```
[root@asterisk ~]# yum install raidtools-1.00.3.tar.gz
```

- Editar el archivo /etc/raidtab quedando de la siguiente forma:

```
raiddev /dev/md0 - - Unidad RAID md0
raid-level 1 - - Escogemos el nivel 1
nr-raid-disks 2 - - Número de discos físicos o particiones del grupo
nr-spare-disks 0 - - Número de discos libres
```

```

chunk-size 4 - - Menor valor de data escrita en discos
persistent-superblock 1
device /dev/sda1
raid-disk 0
device /dev/sdb1
raid-disk 1

```

- Inicializar el dispositivo RAID y formatear el sistema de archivos
 

```
mkraid /dev/md0
mke2fs /dev/md0
```
- En caso se desee detener, iniciar o modificar el RAID se utiliza las siguientes herramientas:
  - raidstart: Inicia un array
  - raidstop: Detiene un array
  - raidhotadd: Se utiliza para añadir discos en pleno funcionamiento (hot swap)
  - raidhotremove: Contrario a lo expuesto anteriormente

Tener en cuenta que estos pasos se deben hacer antes de que procedamos con la instalación de Asterisk y paquetes, ya que al formatear las unidades perderíamos toda la información.

### 3.7. Plan de Marcación

Para desarrollar el tema de numeración se tiene en cuenta varias consideraciones.

#### 3.7.1. Numeración interna

En este caso se separa la numeración de acuerdo al área, cada una de ellas tendrá restricciones de acuerdo a la necesidad de comunicación. Cada uno de ellos tendrá un número de buzón de voz adicional al número de anexo. Los números de anexos serán de 3 dígitos 1XX al 9XX. La operadora o secretaria recepcionista tendrá el número de anexo 100. A continuación se muestra en la tabla 3.5 la numeración interna (anexos) y restricciones por área.

#### 3.7.2. Numeración externa

##### 3.7.2.1. Llamadas entrantes

El proveedor del servicio primario (30 canales de voz) nos asigna un rango de numeración llamado DID (Direct Inward Dialing) que es el rango que puede ser

completado por el servidor Asterisk. Por lo general el rango asignado depende del proveedor, pero se asume que es el rango siguiente: 710505X, entonces se puede completar 10 numeraciones, desde el 7105050 al 7105059. Cuando algún abonado de la RTPC realice una llamada hacia alguno de los DID, el servidor Asterisk pasará a tomar el control de la llamada y la enrutará de acuerdo al plan de numeración.

**Tabla 3.5. Tabla de anexos y restricciones de acuerdo a área**

Área / Posición	Anexo	Restricciones
Operadora	100	801
Gerente General	101	801
Subgerencias	102 - 110	801
Sistemas	111 - 150	801
Operaciones y Mantenimiento	151 - 199	Móviles, LDN, LDI y 0801
Contabilidad	201 - 299	LDI y 0801
Ventas	301 - 399	LDI y 0801
Fax	400	Móviles, LDI y 0801

Se implementará un IVR (Interactive Voice Response) donde el interlocutor escuche una locución de saludo solicitando que marque el número de anexo si es que lo conoce, se le dará el anexo principal de cada área como opción de marcado, se le indicará si desea enviar fax de no recibir algún tono marcado por el interlocutor, se le derivará al anexo de la operadora. Para el caso del fax el número telefónico es 7105060 ya que es una línea analógica independiente, los demás DIDs serán enrutados y traducidos por el Asterisk según el anexo u opción marcada por el abonado llamante.

Otra consideración a tener en cuenta, es la siguiente: el anexo del gerente general no podrá recibir llamadas directas, sino únicamente deben ser transferidas desde el anexo de secretaria de gerencia, salvo que marquen su número directo, para esto se le asignará el número directo DID 7105055.

Adicionalmente en nuestro servidor Asterisk estará activado la función de voicemail (buzón de voz), el cual se adecuarán los parámetros para que active la función de mensajería de voz luego de 5 timbradas y para que el mensaje de voz sea de 30 segundos como máximo.

### 3.7.2.2 Llamadas salientes

El proveedor del servicio primario asigna un DOD (Direct Outward Dialing) que es el número marcado en cualquier llamada saliente originada en nuestros terminales; para este

caso es el 7105050. Las llamadas salientes en gran parte estarán enrutadas vía el acceso primario, de sufrir alguna caída de servicio en el primario, quedará como única alternativa la salida vía la línea analógica de fax y las liceas móviles. El prefijo para las llamadas nacionales es el 0 y se tendrá en cuenta los códigos válidos para las provincias en el Perú, las llamadas LD y LDI serán enrutadas vía el primario, ya que el operador de telefonía ofrece un plan de minutos para este tipo de consumo. El prefijo para llamadas internacionales es el 00, el prefijo para llamadas a móviles locales el 9, aquí se diferenciará las series de numeración del operador Claro ya que se tiene un chip para Claro y dos de Movistar pues los demás operadores móviles (Movistar y Nextel) serán enrutados por la licea de Movistar.

### **3.8. Medidas de seguridad en la red.**

Es conocido que últimamente existen herramientas como los programas SIPCrack y Cain y Abel que hacen posible cometer fraudes en redes SIP. Estas herramientas hacen un barrido en las redes para encontrar algún host que ofrezca servicios SIP, una vez encontrado, realizan un barrido para encontrar extensiones y contraseñas por medio de fuerza bruta o de ataques de palabra de diccionario. En primer lugar el servidor Asterisk está en nuestra red privada y será imposible algún ataque externo, ya que al servidor Asterisk no le damos acceso a Internet. Aún así el ataque puede ser desde dentro de nuestra red; para contrarrestar estos ataques, existen ciertas reglas de seguridad que ya están implementadas en Asterisk y sólo hay que aplicarlas para proteger al servidor de barridos masivos. Las reglas a seguir son las siguientes:

- \* No aceptar pedidos de autenticación de cualquier dirección IP. Para esto se utiliza las líneas "permit" y "deny" del archivo sip.conf. En nuestro caso el archivo sip.conf incluiría las siguientes líneas de comando:

```
permit=172.16.140.0/255.255.255.0
deny=0.0.0.0/0.0.0.0
```

- \* Establecer el valor de la entrada "allowauthreject=yes" en el archivo sip.conf. El valor por defecto es "no" lo que puede ser potencialmente inseguro. Estableciendo este valor en "yes" se rechazarán los pedidos de autenticación fallidos utilizando nombre de extensiones válidas con la misma información de un rechazo de usuario

inexistente. De esta forma no se facilita la tarea al atacante para detectar nombres de extensiones existentes utilizando técnicas de "fuerza bruta".

- \* Establecer la línea `allowguest=no` en el archivo `sip.conf`, esto no permitirá conexiones de agentes SIP sin contraseña.
- \* Utilizar usuarios con nombres distintos al número de las extensiones y claves seguras para cada usuario SIP. En este caso utilizaré claves que combinen letras mayúsculas, minúsculas y números. Por ejemplo para la extensión SIP 100, el usuario será `4n3xo100` y le corresponderá la clave `*.Empr3sA100#` ; al anexo 101 le corresponde el usuario `4n3xo101` y la clave `*.EmpresA101` ; así sucesivamente con todas las extensiones SIP, esto se notará en el archivo `sip.conf` que se presentará luego.
- \* Bloquear los puertos del Asterisk Manager Interface configurando las siguientes líneas en el archivo `manager.conf`:
 

```
[admin]
secret=*.Adm1N#
permit=172.16.1.140/255.255.255.0
permit=127.0.0.1/255.255.255.0
deny=0.0.0.0/0.0.0.0
```

### 3.9. Puesta en servicio y configuración de los archivos del servidor Asterisk

#### 3.9.1. Canales físicos

Archivo `/etc/daahdi/system.conf`

```
// Canales del acceso primario
span=1,0,0,ccs,hdb3
bchan=1-15
dchan=16
bchan=17-31

// Canales FXO y FXS
fxoks=32-35 // La línea analoga de fax llega a un puerto FXO y dos FXO van hacia las
liceas. El canal 35 es la línea fax que llega desde la RTPC, el 32 va a la licea de Claro y
el 33 y 34 a las liceas de Movistar
fxsks=36-39 // Un puerto FXS va hacia el equipo fax
defaulttone=es
```

```
loadzone=es
echocanceller=mg2,1-15,17-34,36-38 // A los canales 35 y 39 no se le aplica
cancelación de eco porque este canal es el fax
```

### 3.9.2. Canales lógicos y señalización

Archivo /etc/dahdi/chan\_dahdi.conf

```
// configuración global de los canales

[channels]
context=default
usecallerid=yes
hidecallerid=no
callwaiting=yes
callwaitingcallerid=yes
threewaycalling=yes
transfer=yes
canpark=yes
cancallforward=yes
callreturn=yes
echocancel=yes
relaxdtmf=yes // parámetro que ayuda a la detección de tonos DTMF
rxgain=0.0
txgain=0.0
group=1
callgroup=1
pickupgroup=1
immediate=no

// Configuración de los canales del primario
switchtype=euroisdn - - protocolo usado por los proveedores
context=from_pstn
group=1
echocancel=yes
signalling=pri_cpe - - señalización donde fuente de clock es la central office
channel =>1-15,17-31 - - el canal 0 es de sincronismo. el 16 es de señalización

// Configuración de los canales FXO
context=from_fxo
signalling=fxs_ks
answeronpolarityswitch=yes
group=2
channel=> 32 - - conectado a la licea de Claro
context=to_licea
signalling=fxs_ks
answeronpolarityswitch=yes
group=3
channel=> 33-34 - - conectado a las liceas de Movistar
```

```

context=to_licea
signalling=fxs_ks
answeronpolarityswitch=yes
group=4
channel=> 35

```

// Configuración de los canales FXS

```

context=from_fxs
signalling=fxo_ks
answeronpolarityswitch=yes
faxdetect=both - - se le indica que detecte tono de fax en los dos sentidos
group=1 - - el canal FXS pertenece al mismo grupo 1 ya que es un canal de desborde
en caso no haya disponibilidad de ninguno de los 30 canales del acceso primario.
channel=> 36-39

```

### 3.9.3. Usuarios y anexos SIP

Para una mejor organización se crea un archivo “sip\_area.conf” para cada una de las áreas que tenemos en la empresa, al final las incluimos a todas en el archivo general sip.conf. Esto es aplicando el concepto de inclusión en los contextos.

Así tenemos:

Archivo sip.conf

; Contexto general

[general]

port = 5060

language=es

bindaddr=0.0.0.0

disallow=all

allow=ulaw

allow=alaw

allow=g729

allow=g723

#include sip\_gerencia.conf

#include sip\_ventas.conf

#include sip\_sistemas.conf

#include sip\_operaciones.conf

#include sip\_contabilidad.conf

Para cada uno de los archivos de configuración por área, solo se muestra el primer usuario, los demás usuarios tienen el mismo formato. El parámetro secret es la contraseña. Tener en cuenta que el usuario, la contraseña y la IP son los datos que tienen que configurarse en el teléfono IP, así como la IP del servidor SIP, para su correcto registro de cada usuario en el servidor Asterisk.

## Archivo sip\_gerencia.conf

```
[100]
type=friend
secret=*.Empr3sA100#
host=dynamic
defaultip=172.16.140.2
callgroup=2
pickupgroup=2
context=gerencia
mailbox=100@default
```

## Archivo sip\_ventas.conf

```
[301]
type=friend
secret=*.Empr3sA301#
host=dynamic
defaultip=172.16.140.11
callgroup=2
pickupgroup=2
context=ventas
mailbox=301@default
```

## Archivo sip\_sistemas.conf

```
[111]
type=friend
secret=*.Empr3sA111#
host=dynamic
defaultip=172.16.140.51
callgroup=2
pickupgroup=2
context=sistemas
mailbox=111@default
```

## Archivo sip\_operaciones.conf

```
[151]
type=friend
secret=*.Empr3sA151#
host=dynamic
defaultip=172.16.140.71
callgroup=-
pickupgroup=-
```

```
context=operaciones
mailbox=151@default
```

Archivo sip\_contabilidad.conf

```
[201]
type=friend
secret=*.Empr3sA201#
host=dynamic
defaultip=172.16.140.31
callgroup=2
pickupgroup=2
context=contabilidad
mailbox=201@default
```

### 3.9.4. Configurando el Buzón de Voz

Aquí se describe el primer usuario de cada área, los demás usuarios se configuran de forma similar. El password es el mismo número de usuario. Este password será solicitado cuando se desee escuchar los mensajes del buzón de voz. Para rescatar los mensajes dejados en el buzón de voz se implementara un contexto funciones en el archivo extensions.conf que se verá con detalle más adelante.

Archivo /etc/asterisk/voicemail.conf

```
[default]
maxmsg=50 - - define como máximo 50 mensajes
maxsecs=30 - - define como máximo mensajes de voz de 30 segundo
100 = 100, Juan Perez,jperez@hotmail.com
301 => 301, Carlos Juarez,cjuarez@hotmail.com
111 => 111, Juana Bello,jbello@hotmail.com
151 = 151, Mario Kempes,mkempes@hotmail.com
201 = _01, Bertha Sanchez,bsanchez@hotmail.com
```

### 3.9.5. Configuración del plan de marcación

Este es el archivo principal que tendrá la descripción del flujo de llamadas tanto entrantes, salientes entre anexos.

**Archivo extensions.conf**

//Llamadas hacia y entre anexos SIP:

```
[anexos]
exten => _[1-3]XX,1,Dial(SIP/${EXTEN},30,Tt)
exten => _[1-3]XX,2,VoiceMailMain(${EXTEN}@default)
exten => _[1-4]XX,3,Hangup
exten => 400,1,Dial(DAHDI/39)
exten => 400,2,Hangup
```

//Llamadas locales salientes

```
[destinoslocales]
exten => _NXXXXXXX,1,Dial(DAHDI/g1/${EXTEN})
exten => _NXXXXXXX,2,Hangup
```

//Destinos 0800 salientes

```
[destinos0800]
exten => _0800.,1,Dial(DAHDI/g1/${EXTEN})
exten => _0800.,2,Hangup
```

//Destinos LDN salientes

```
[destinosldn]
exten => _0[4-8].,1,Dial(DAHDI/g1/${EXTEN})
exten => _0[4-8].,2,Hangup
```

//Destinos LDI salientes

```
[destinosldi]
exten => _00.,1,Dial(DAHDI/g1/${EXTEN})
exten => _00.,2,Hangup
```

//Destinos Móviles. Aquí se distingue las series locales del operador Claro, esta información actualizada se obtiene de la web del MTC.

```
exten => _9XXXXXXXX,1,Dial(DAHDI/g4/${EXTEN})
exten => _9XXXXXXXX,2,Hangup
exten => _991XXXXXXXX,1,Dial(DAHDI/g3/${EXTEN})
exten => _991XXXXXXXX,2,Hangup
exten => _992XXXXXXXX,1,Dial(DAHDI/g3/${EXTEN})
exten => _992XXXXXXXX,2,Hangup
exten => _993XXXXXXXX,1,Dial(DAHDI/g3/${EXTEN})
exten => _993XXXXXXXX,2,Hangup
exten => _9943XXXXXXXX,1,Dial(DAHDI/g3/${EXTEN})
exten => _9943XXXXXXXX,2,Hangup
exten => _9946XXXXXXXX,1,Dial(DAHDI/g3/${EXTEN})
exten => _9946XXXXXXXX,2,Hangup
```

```

exten => _9947XXXXXX,1,Dial(DAHDI/g3/${EXTEN})
exten => _9947XXXXXX,2,Hangup
exten => _9948XXXXXX,1,Dial(DAHDI/g3/${EXTEN})
exten => _9948XXXXXX,2,Hangup
exten => _9949XXXXXX,1,Dial(DAHDI/g3/${EXTEN})
exten => _9949XXXXXX,2,Hangup
exten => _997XXXXXX,1,Dial(DAHDI/g3/${EXTEN})
exten => _997XXXXXX,2,Hangup

```

**//Configurando las restricciones.** Aquí se incluye los contextos de acuerdo a restricciones por área

#### [gerencia]

```

include => anexos
include => destinoslocales
include => destinosmoviles
include => destinosldn
include => destinosldi
include => destinos0800
include => funciones

```

#### [sistemas]

```

include => anexos
include => destinoslocales
include => destinosmoviles
include => destinosldn
include => destinosldi
include => destinos0800
include => funciones

```

#### [operaciones]

```

include => anexos
include => destinoslocales
include => destinos0800
include => funciones

```

#### [contabilidad]

```

include => anexos
include => destinoslocales
include => destinosmoviles
include => destinosldn
include => destinos0800
include => funciones

```

#### [ventas]

```

include => anexos
include => destinoslocales
include => destinosmoviles
include => destinosldn

```

```
include => destinos0800
include => funciones
```

```
[from_fxs]
include => anexos
include => destinoslocales
include => destinosldn
include => destinos0800
include => funciones
```

### **//Configurando el contexto [funciones] para implementar el buzón de voz**

Para escuchar sus mensajes de voz, los usuarios deben marcar 777. Escucharan la locución de bienvenida y el sistema solicitara su clave que en este caso es igual al numero de usuario o número de anexo.

```
[funciones]
exten => 777,1,Answer()
exten => 777,n,VoiceMailMain(@default)
```

### **//Configurando las llamadas entrantes por el primario**

Aquí se le ofrecerá al usuario un menú IVR de bienvenida, donde se le indicaran los anexos principales de cada área, el anexo del fax, si el usuario no llegara a marcar ningún dígito en 5 segundos, la llamada será desviada a la operadora. También se tiene en cuenta que para las llamadas desde la RTPC no pueden contactar directamente con el anexo de gerencia a menos que llamen al número directo, en primer lugar se le deriva a la operadora y esta ya podría transferir la llamada al anexo 101 de gerencia a través del contexto [anexossip].

```
[from_pstn]
exten => s,1,Answer()
exten => s,n,Background(bienvenida_opciones)
exten => s,DigitTimeout,5
exten => s,n,Dial(SIP/100,30,r)
exten => s,n,Hangup()
exten => _[1-3]XX,1,Goto(anexos)
exten => 101,1,Dial(SIP/100)
exten => 101,2, Hangup()
exten => 400,1,Dial(DAHDI/39)
exten => 400,2, Hangup()
exten => 7105055,1,Dial(SIP/101)
exten => 7105055,2, Hangup()
```

Al final se tiene listo los archivos de configuración del servidor para su puesta en servicio.

### 3.10. Monitoreo de llamadas en el sistema Asterisk

Un punto muy importante en cualquier sistema de telefonía es la gestión de tráfico y monitoreo de las llamadas realizadas, sobre todo para las llamadas salientes a fin de evitar el uso indebido de las comunicaciones de voz que generaría pérdidas económicas y de productividad en la empresa.

Asterisk por defecto genera un archivo de registro (CDR : Call Detail Record) de todas las llamadas realizadas por los distintos agentes, el cual se almacena en el directorio `/var/log/asterisk/cdr-csv` y el nombre del archivo es `Master.csv` que es un archivo de texto cuyos datos están separados por comas. Este archivo puede ser exportado a Excel o a una base de datos para su análisis respectivo. Para que Asterisk guarde estos CDRs, es necesario verificar que el módulo `cdr_csv.so` esté activo, esto se hace desde la consola de comandos con el “show modules” y con “cdr status”. Los campos del archivo se muestran en la tabla 3.6.

**Tabla 3.6. Descripción de campos del archivo Master.csv**

<b>CAMPO</b>	<b>DESCRIPCION</b>
<b>accountcode</b>	Código de cuenta, consta de 20 caracteres como máximo.
<b>src</b>	Número de Caller*ID, máximo 80 caracteres
<b>dst</b>	Extensión destino, máximo 80 caracteres
<b>dcontext</b>	Contexto destino, máximo 80 caracteres
<b>clid</b>	Caller*ID con texto, máximo 80 caracteres
<b>channel</b>	Canal usado, máximo 80 caracteres
<b>dstchannel</b>	Canal destino, máximo 80 caracteres
<b>lastapp</b>	Última aplicación, máximo 80 caracteres
<b>lastdata</b>	Argumento usado en la última aplicación, máximo 80 caracteres
<b>start</b>	Fecha y hora de inicio de llamada
<b>answer</b>	Fecha y hora en que se contestó la llamada
<b>end</b>	Fecha y hora de fin de la llamada
<b>duration</b>	Duración de la llamada en segundos
<b>billsec</b>	Tiempo total de la llamada
<b>disposition</b>	Indica lo acontecido con la llamada, valores: ANSWERED, NO ANSWER, BUSY, FAILED
<b>amaflags</b>	Tipo de señalización utilizada, valores: DEFAULT, DOCUMENTATION, BILL u OMIT
<b>uniqueid</b>	Identificador único de la llamada

Para hacer un reporte de tráfico se puede exportar semanalmente a una base de datos o a una hoja de Excel, cargando el archivo de texto delimitando las columnas por la coma.

Cuando se trata como archivo de texto, está de más preocuparse por el tamaño en bytes que ocupa éste archivo en el sistema, ya que semanalmente se puede eliminar y crear un nuevo archivo.

Como alternativa, existen herramientas de monitoreo de software libre como Astbill, Areski Stat v2 o A2Billing que sirven para análisis y reportes de tráfico y que interactúan con la base de datos de Asterisk.

## CONCLUSIONES

- 1.- Al culminar el presente informe se observa claramente la ventaja económica que otorga el implementar un sistema de telefonía con software libre Asterisk y el hardware que implica su arquitectura en comparación con la adquisición de una central PBX y demás hardware de algún fabricante propietario.
- 2.- En este informe se ha presentado solo algunas bondades que ofrece Asterisk, ya que como lenguaje es una herramienta de desarrollo que puede integrarse fácilmente con los demás lenguajes de programación, base de datos, etc., permitiendo diseñar alguna solución ante cualquier requerimiento que implique mayor nivel de complejidad.
- 3.- Se observa la facilidad de aprendizaje de los conceptos básicos del lenguaje Asterisk que es de alto nivel; siendo suficiente estos conocimientos para administrar, gestionar y desarrollar nuestras propias centrales con diversos servicios de comunicación. Es requerido no obstante conocimientos básicos sobre el sistema operativo Linux, conceptos de telefonía analógica y digital y algo de programación.
- 4.- La solución para una empresa con menos de 100 trabajadores y una sola sucursal no necesita de hardware o software para garantizar calidad de servicio ni priorización de tráfico, ya que el rendimiento de la red Ethernet no llega a afectar las comunicaciones de voz debido al vasto ancho de banda que oscila entre los 10/100/1000Mbps el poco retardo que experimentan estas redes. En el caso de PYMES con sucursales en distintas locaciones no aplicaría esta conclusión, ya que en este caso si es necesario garantizar un ancho de banda para la voz a nivel WAN en el enlace de datos.

- 5.- Como la tasa de bits no es problema en la red, se recomienda hacer el uso del codec G711 ya que tiene el mejor rendimiento en cuanto a calidad de la comunicación.
- 6.- En cuanto a hardware, la recomendación es trabajar de acuerdo a los tipos de teléfonos con los que cuenta la empresa. Es decir si la empresa cuenta con teléfonos analógicos, la sugerencia es utilizar dispositivos ATA en vez de adquirir teléfonos IP. La solución presentada en este informe es para una empresa que recién empieza las actividades y que solo tiene implementado la red Ethernet.
- 7.- Para el caso de lenguajes y software libre, hay numerosos aportes e información en Internet, de la cual uno puede valerse para su propio conocimiento o para contribuir con el desarrollo del mismo software.
- 8.- En la actualidad en la telefonía IP, SIP esta desplazando al antiguo H323 por su rapidez y modularidad, y porque el protocolo SIP nació para establecer comunicaciones de video y audio sobre Internet.
- 9.- El archivo principal de configuración es el “extensions.conf” ya que es el que describe la lógica del enrutamiento y flujo de las llamadas en todo el sistema, para lo cual se dispone de herramientas de programación, la estabilidad y eficiencia dependerá mucho de la lógica y del nivel de abstracción con el que se planteo este plan de marcación.

**ANEXO A**  
**GLOSARIO DE TÉRMINOS**

**ADSL:** Asymmetric Digital Subscriber Line

**AGI:** Asterisk Gateway Interface

**ATA:** Analog Telephone Adaptor

**CLC:** Close Logical Channel. Mensaje de H245 para cerrar un canal lógico y finalizar una llamada

**CNAME:** Canonical Name. Registro en base de datos DNS, registra un alias de algún host

**CODEC:** Codificador – Decodificador

**CO:** Central Office

**CPU:** Central Processing Unit

**Dahdi:** Digium Asterisk Hardware Device Interface. Driver para controlar hardware de telefonía

**DCF:** Disengage Request Confirm. Confirmación a DRQ

**DID:** Direct Inward Dialing

**DOS:** Disk Operating System

**DOD:** Direct Outward Dialing

**DRQ:** Disengage Request. Mensaje RAS H225 solicitando finalización de llamada

**DRJ:** Disengage Request Reject. Rechazo a DRQ

**DSP:** Digital Signal Processor

**DTMF:** Dual Tone Multi-Frequency

**E1:** Arreglo de 32 tramas, 2 de sincronismo y 30 de datos, cada una de 8 bits en un tiempo de 125 useg

**FXS:** Foreign Exchange Station

**FXO:** Foreign Exchange Office

**G711:** Codec de voz cuya tasa de bits es 64 kbps. También es llamado PCM

**G729:** Codec de voz con compresión y algoritmo licenciado cuya tasa es de 8 kbps

**GNU:** Not Unix

**GSM:** Codec de voz usado en tecnología GSM, tasa de bits es 13 kbps

**H225:** Protocolo de la pila de H323, usado para la señalización en comunicaciones de voz sobre IP

**H245:** Protocolo de la pila de H323, usado para control de canal en comunicaciones de voz sobre IP

**H323:** Conjunto de estándares y recomendaciones de la ITU-T para realizar comunicaciones audiovisuales sobre IP

**HUNG UP:** Colgar el auricular

**HTTP:** Hypertext Transfer Protocol

**IAX:** Inter-Asterisk Exchange Protocol

**IETF:** Internet Engineering Task Force

**IP:** Internet Protocol. Dirección lógica y jerárquica de un dispositivo de red

**ISDN:** Integrated Services Digital Network

**IVR:** Interactive Voice Response

**MGCP:** Media Gateway Control Protocol

**MOS:** Mean Opinión Score

**Off-Hook:** Descolgar el teléfono. Impedancia despreciable

**OLC:** Open Logical Channel. Mensaje de H245 para abrir un canal lógico.

**On-Hook:** Colgar el teléfono. Impedancia alta

**PBX:** Private Branch Exchange

**PCI:** Peripheral Component Interconnect. Bus de ordenador.

**PCI-Express:** Peripheral Component Interconnect Express. Reemplaza a PCI, PCI-X y AGP.

**PCMA:** Variante del codec G711 con ley A

**PCMu:** Variante del codec G711 con Ley u

**PCM:** Pulse Code Modulation **PSTN:** Public Switched Telephone Network

**PYME:** Pequeña y Mediana Empresa

**Q931:** Protocolo de control de conexiones ISDN. Establece, mantiene y libera conexiones entre dos DTE a través del canal D

**RAM:** Random Access Memory

**RAS:** Registration Admission and Status

**RDSI:** Red Digital de Servicios Integrados, en inglés ISDN

**RPE-LTP:** Regular Pulse Excited-Long Term Prediction, nombre del codec GSM

**RTPC:** Red de Telefonía Pública Conmutada, en inglés PSTN

**RTP:** Real-Time Transport Protocol definido en el RFC 3550

**RTCP:** Real-Time Control Transport Protocol definido en el RFC 3550

**SATA:** Serial ATA, sistema controlador de discos reemplaza al IDE/ATA

**SDP:** Session Description Protocol

**SETUP:** Mensaje del protocolo Q931 para iniciar una llamada

**SIP:** Session Initial Protocol protocolo de inicio de sesiones para multimedia

**SMTP:** Simple Mail Transfer Protocol

**STREAM:** Trama de Datos de una determinada cantidad de bits

**TCS:** Terminal Capability Set, mensaje H245 para negociar capacidades entre los terminales VoIP

**TDM:** Time Division Multiplexing

**TR:** Terminal Remota

**UAC:** Agente de Usuario Cliente

**UAS:** Agente de Usuario Servidor

**UDP:** User Datagram Protocol

**UIT:** Union Internacional de Telecomunicaciones

**UNIX:** Sistema Operativo multitarea y multiusuario

**VoIP:** Voice Over IP, conjunto de estándares que hacen posible la comunicación de voz sobre el protocolo TCP/IP

**VLAN:** Virtual LAN, protocolo estándar 802.1Q usado para crear redes lógicas

**Vrms:** Voltaje eficaz

**WAN:** Wide Area Network

**WDM:** Wavelength Division Multiplexing

**WLL:** Wireless Local Loop

**Zaptel:** Nombre de driver usado en Asterisk hasta el 2007, creado por Jim Dixon para control de hardware de telefonía

## BIBLIOGRAFIA

- 1.- Andrew S. Tanenbaum, “Redes de Computadoras”, Pearson Educación – México, Cuarta Edición, 1999
- 2.- Leon W. Couch II, “Sistemas de Comunicación Digitales y Analógicos”, Prentice Hall, México, Quinta edición, 1998
- 3.- Alfredo Certain Yance, “Asterisk Comunicaciones de Código Abierto”, Gecko Networks – Colombia, pre-edición 2007
- 4.- Instituto LOGTEL, “Telefonía IP, Componentes, Protocolos y Servicios”, Chile, 2007
- 5.- Daniel Pizarro, “Manual del Curso de Asterisk”, Infobox Latinoamérica sucursal Perú, 2008
- 6.- David Gomillion, Barrie Dempster, “Construyendo Sistemas Telefónicos con Asterisk”, Pack Publishing, 2007
- 7.- Instituto TECSUP, “Centrales Telefónicas y Servicios de Voz”, Programa de Capacitación Continua 2006
- 8.- Instituto TECSUP, “Telefonía sobre redes IP”, Programación de Capacitación Continua 2006
- 9.- José Manuel Huidobro, David Roldan, “Tecnología VoIP y tecnología IP”, Creaciones Copyright, 2006
- 10.- Jim Van Meggelen, Leif Madsen, “Asterisk: The Future of Telephony”, O’Reilly, Segunda edición, 2007
- 11.- Cotización enviada por empresa Convexus para solución con PBX Nortel BCM-400 actualizada a la fecha de abril 2009.
- 12.- Cotización con precios referenciales de equipos de telefonía IP enviado por empresa Sumtec actualizado a la fecha mayo 2009
- 13.- Web [www.voip-info.org](http://www.voip-info.org)
- 14.- Web [www.digium.com](http://www.digium.com)
- 15.- Web [www.asterisk.org](http://www.asterisk.org)