

**UNIVERSIDAD DON BOSCO
FACULTAD DE INGENIERÍA**



**TRABAJO DE GRADUACIÓN
PARA OPTAR AL GRADO DE
INGENIERO EN CIENCIAS DE LA COMPUTACIÓN**

**ESTUDIO MONOGRÁFICO DEL PROTOCOLO SIP Y SU APLICACIÓN EN
SERVICIOS DE VOZ SOBRE IP**

PRESENTADO POR
**Melvin Rodrigo Castro Polanco
Sandra Lissette Hernández Alvarenga**

ASESOR
Carlos Guillermo Bran

SEPTIEMBRE 2008
EL SALVADOR, CENTRO AMÉRICA

UNIVERSIDAD DON BOSCO
FACULTAD DE INGENIERIA



RECTOR
ING. FEDERICO MIGUEL HUGUET RIVERA

SECRETARIO GENERAL
ING. XIOMARA YESENIA MARTÍNEZ

DECANO FACULTAD DE INGENIERIA
ING. ERNESTO GODOFREDO GIRON

SEPTIEMBRE 2008
EL SALVADOR, CENTRO AMÉRICA

UNIVERSIDAD DON BOSCO
FACULTAD DE INGENIERIA



**TRABAJO DE GRADUACION PARA OPTAR AL GRADO DE
INGENIERO EN CIENCIAS DE LA COMPUTACION**

**ESTUDIO MONOGRÁFICO DEL PROTOCOLO SIP Y SU APLICACIÓN EN
SERVICIOS DE VOZ SOBRE IP**

ASESOR

LECTOR

SEPTIEMBRE 2008
EL SALVADOR, CENTRO AMÉRICA

Prefacio

Desde siempre la comunicación ha formado parte elemental en la vida del ser humano, permitiéndole a la tecnología basar parte de su desarrollo a este elemento. Es así como surge la tecnología Voz sobre IP, que ha venido gestionando una nueva forma de tecnología orientada a las centrales telefónicas tradicionales y al transporte de datos a través de internet; Esta nueva tendencia se menciona como integración de voz y datos, y busca entre otras cosas, unificar ambos servicios en una sola red; pero esta tecnología requiere utilizar protocolos, tales como los protocolos encargados del establecimiento de la comunicación, los cuales son importante conocer a fondo para lograr obtener todas las ventaja de la tecnología Voz sobre IP.

Es por esta razón que la investigación presente se centra en el estudio del protocolo SIP, detallando aspectos importantes, tales como: historia, funcionamiento, rol que toma en la realización de la comunicación telefónica, problemas que presenta actualmente y posibles soluciones. Por otro lado cubre las características del funcionamiento y configuración de una central telefónica (PBX) a partir de esta tecnología, con el uso y configuración del software libre Asterisk utilizando el protocolo SIP.

Agradecimientos

Muchos han sido los días llenos de dedicación y esfuerzo que me han llevado a concluir esta etapa tan importante de mi vida. Etapa que jamás hubiera sido posible sin el aporte de cada uno de los seres que durante este trayecto estuvieron a mi lado.

Quiero iniciar mi agradecimientos con el ser principal que nunca me abandono, que me demostró que la lucha sigue y jamás me dejo vencer, ni alejo la luz de la esperanza en mi camino, me dirijo a nuestro omnipotente Dios que siempre me ilumino y puso en mi camino a las mejores personas que me hicieron crecer no solo profesionalmente si no también humanamente.

Continuo mi agradecimiento, dirigiéndome a las dos personas que sin ellas está meta jamás hubiera sido posible, mis padres Víctor Hernández y Rhina Alvarenga, hacia quienes es difícil encontrar las suficientes palabras que reflejen el enorme agradecimiento que siento ante todo el esfuerzo y la constate lucha que realizaron día con día, con la esperanza y el deseo que este día llegara, Muchas Gracias Papas...

A mis hermanas Glenda Hernández y Ana María Hernández, por su apoyo, comprensión y fé en que esta meta se realizaría.

A mi novio pero sobre todo mi amigo, Carlos Mauricio Castillo, en quien siempre encontré la comprensión indispensable, el apoyo incondicional y el cariño, que me dieron fuerzas para no dejarme vencer y continuar en la lucha.

A mis amigos a quienes no podría dejar de mencionar, porque siempre estuvieron apoyándome y brindándome su ayuda en cualquier momento; Gracias Juan José Ventura, Gabriel Ruiz, Karen Cea, Catherine Santana, Enrique Castillo, Verónica Villegas, Rodrigo Rubio, Katleen Lemus, Blanca Méndez y Rodrigo Castro con quien también luchamos juntos en este proyecto.

Gracias a todos los maestros que me aportaron parte de sus conocimientos, a nuestro asesor de tesis por guiarnos, compañeros de trabajo que me apoyaron y todas aquellas personas que fueron parte de esta lucha.

A Todos muchas gracias....

Sandra Lissette Hernández Alvarenga

Luego de alcanzar esta meta tan importante en mi vida, ante todo, doy gracias a Dios y a la Virgen María, por permitirme día a día dar los pasos que me han llevado a lograr este objetivo que inicie hace más de cinco años.

Gracias a las personas más importantes en mí vida y a quienes debo este éxito y más de lo que alguna vez podre alcanzar: mi padre, Jorge Alberto Castro que está en el cielo, y mi mamá, Armida Emely Polanco de Castro, ya que sin el esfuerzo, dedicación, apoyo pero sobre todo el amor que me han brindado durante todos los años de mi vida, y principalmente durante estos últimos, no hubiera podido alcanzar este éxito, que es suyo.

A mis hermanas, Meybell Emely Castro y Karen Estefanía Castro, quienes a pesar de no entenderme en algunos casos, y sin importar las circunstancias, no han dejado de creer en mí y han estado ahí para brindarme su apoyo.

Gracias a esa persona especial que Dios ha puesto en mi camino: Ana Celina Arévalo; quien en más de una ocasión, con algún regaño o consejo, pero sobretodo con su cariño y apoyo, me ha ayudado a fijarme metas y alcanzarlas.

A mis amigos y mi familia en general, que me han brindado su apoyo y comprensión desde un inicio; y de manera muy especial a Sandra Lissette Hernández por decidirse a emprender este camino.

A todos mis maestros y nuestro asesor Carlos Bran por dedicar su tiempo a este proyecto, y a todas las personas que de alguna y otra forma me han acompañado en el desarrollo académico y profesional.

Muchas Gracias.

Melvin Rodrigo Castro Polanco

Índice

Introducción	i
Capítulo I MARCO REFERENCIAL	
1.1 Antecedentes	2
1.2 Importancia de La Investigación	3
1.3 Planteamiento del Problema.	4
1.4 Definición del Tema	5
Descripción General del Tema	5
1.5 Justificación	6
1.6 Objetivos	7
1.6.1 Objetivo General.	7
1.6.2 Objetivos Específicos	7
1.7 Alcances	8
1.8 Limitaciones	9
1.9 Delimitación.	10
1.10 Marco Teórico	11
1.10.1 Referencias Históricas	11
1.10.2 Marco Conceptual.	13
Red de Telefonía Pública.	13
Tecnología VoIP.	15
Asterisk	21
1.10.3 Marco Experimental.	21
1.11 Metodología De La Investigación	23
Recopilación de La Información	23
Documentación y Análisis.	24
Implementación.	24
1.12 Plan De Acción Y Cronograma	25
1.12.1 Plan de Acción	25
1.12.2 Cronograma	29
Diagrama de Gantt	29
1.13 Plan Capitular	32

1.14 Presupuesto	37
1.14.1 Inversión	37
Capítulo II INTRODUCCIÓN A LA TECNOLOGÍA VoIP	
Introducción	40
2.1 Generalidades de la Telefonía	41
2.1.1 Evolución Histórica de la Telefonía	40
2.1.2 Funcionamiento de la Telefonía Actual.	43
2.1.3 Características de la Telefonía.	47
2.2 Descripción de La Tecnología IP	48.
2.2.1 Historia Y Generalidades de Las Tecnologías De VoIP	49
2.2.2 Funcionamiento de Los Sistemas VoIP	49
2.2.2.1 Proceso de Codificación De Voz	50
2.2.2.2 Códec Utilizados en VoIP	53
▪ G.711	54
▪ G.723	56
▪ G.729	57
2.2.3 Protocolos de Soporte de VoIP	58
2.2.3.1 Protocolos de Señalización	58
▪ H.323	58
▪ SIP	61
▪ Inter Asterisk Exchange AIX2	63
2.2.3.2 Protocolos de Transporte	63
Real Time Protocol RTP	64
Real Time Streaming Protocol RTSP	65
Real Time Control Protocol RTCP	65
2.2.4 Tecnologías Comerciales Existentes de VoIP	65
2.2.5 Características y Limitaciones de La VoIP	66
Capítulo III PROTOCOLO SIP	
Introducción	71
3.0 Protocolo SIP	72

3.1	Historia de SIP	72
3.2	Funcionalidades de SIP	73
	3.2.1 Establecimiento, Modificación y Terminación de Sesión.	73
	3.2.2 Movilidad del Usuario.	73
3.3	Elementos SIP	75
	3.3.1 Agentes de Usuario (UA)	75
	3.3.2 Servidor SIP	76
	3.3.2.1 Proxy SIP	76
	3.3.2.2 Servidor de Localización	79
	3.3.2.3 Servidor de Re Direccionamiento SIP	78
	3.3.2.4 Servidor de Registro	80
3.4	Operación de SIP	81
	3.4.1 Mensajes SIP.	81
	3.4.1.1 Mensaje de Solicitud.	82
	▪ Método.	82
	▪ Invite.	82
	▪ Register	83
	▪ Bye	83
	▪ Ack	84
	▪ Cancel	84
	▪ Options	85
	▪ Refer	85
	▪ Sucrite	86
	▪ Notify	87
	▪ Message.	87
	▪ Info.	88
	▪ Prack	89
	▪ Update	89
	▪ SIP y Sips Uris.	90
	3.4.1.2 Mensaje De Respuesta	91
	• 1xx Mensajes Informativo (Informational)	91
	• 2xx Mensajes de Éxito (Success)	93

• 3xx Mensajes de Redirección (Redirection)	93
• 4xx Mensajes de Error por Parte del Cliente	95
• 5xx Mensajes de Error por Parte del Servidor	99
• 6xx Mensaje de Falla Global (Global Failure)	100
3.4.1.3 Cabecera de Mensaje SIP	101
3.4.1.4 Cuerpo de Mensaje SIP	106
3.4.2 Autenticación.	111
3.4.3 Dificultades Con Los Procesos de Traducción de Direcciones Nat	112
3.4.3 .1 Protocolos En Tiempo Real y Pat	115
3.4.3.2 Esquema de la red SIP y Pat	113
3.4.3.3 Posibles Soluciones a Los Problemas de SIP Con Pat.	115
3.4.4 Operación De SIP En La Capa de Transporte	117
3.4.4.1 Transacción Invite	118
3.4.4.2 Transacción Cancel	118
3.4.4.3 Otras Transacciones	119
3.5 Aplicaciones de SIP	119
3.6 Ejemplo de Comunicación SIP	120
Capítulo IV SOLUCIÓN DE VoIP UTILIZANDO ASTERISK	
Introducción	132
4.1 Definición y Generalidades	133
4.1.1 Private Branch Exchange - PBX.	133
4.1.1.1 Esquema Conceptual	133
4.1.1.2 Aspectos Generales	134
4.1.1.3 Estructura de las PBX	134
4.1.2 Asterisk Como PBX	135
4.1.2.1 Esquema Conceptual	135
4.1.2.2 Aspectos Generales	135
4.2 Historia de Asterisk	136
4.3 Características de Asterisk	137
4.3.1 Característica Generales	137
4.3.2 Licenciamiento de Asterisk	137

4.3.3	Funcionalidades de Llamada	139
4.3.4	Ventajas	140
4.4	Arquitectura de Asterisk	143
4.4.1	Directorios Utilizados por Asterisk	143
4.4.2	Esquema de La Arquitectura de Asterisk	147
4.4.2.1	Canales	147
4.4.2.2	Codecs	148
4.4.2.3	Formatos	149
4.4.2.5	Aplicaciones y Funciones	149
4.4.2.5	Núcleo PBX	150
4.4.3	Dial Plan	150
4.4.4	Función Dial()	156
4.4.5	Limitaciones de La Arquiteura De Asterisk	158
4.5	Asterisk y La Telefonía Tradicional	158
4.5.1	Asterisk y La Telefonía Tradicional: Líneas Analógicas	159
4.5.2	Asterisk y La Telefonía Tradicional: Líneas Analógicas II	160
4.6	VoIP y Asterisk	160
4.7	Utilización de Asterisk A Través de SIP	161
4.8	Usos y Aplicaciones de Asterisk	162
4.8.1	Usos	162
4.8.2	Aplicaciones	164

Capítulo V: GUÍA DE IMPLEMENTACIÓN DE UN SERVIDOR ASTERISK COMO UNA SOLUCIÓN DE VOZ SOBRE IP

5.1	Elementos a Utilizar	166
5.2	Requerimientos Mínimos de Hardware y Software.	166
5.3	Desarrollo de La Guía.	167
Paso 1.	Instalación de Asterisk	167
Paso 2.	Configuración de Asterisk Vía Consola	176
Paso 3.	Configuración de Los Clientes SIP	180
5.4	Establecimiento de La Llamada Asterisk	186

Conclusiones	187
Recomendaciones	189
Fuentes de Información	190
Glosario	192
Anexos	197

Índice de Tablas

Tabla 1 Plan de acción de la ejecución del proyecto.	28
Tabla 2 Cronograma de la ejecución del proyecto.	31
Tabla 3 Honorarios de la persona a desarrollar el proyecto	37
Tabla 4 Papelería y útiles a utilizar en el proyecto	37
Tabla 5 Otros gastos considerados en el desarrollo del proyecto.	38
Tabla 6 Gasto de Equipo para el desarrollo del proyecto	38
Tabla 7 Gasto de Software.	38
Tabla 8 Partes del Mensaje SIP	80
Tabla 9 Los diferentes campos de los 4 tipos de cabeceras del mensaje SIP	102
Tabla 10 Campos del cuerpo del mensaje SIP	107
Tabla 11: Resumen de los principales archivos de Asterisk.	146

Índice de Imágenes

Figura 1: Proceso de codificación de la Voz	50
Figura 2: Arquitectura del protocolo H.323	60
Figura 3: Arquitectura del protocolo SIP	63
Figura 4: Campos del protocolo RTP	64
Figura 5: Interacción de los diferentes Servidores SIP y los AU	76
Figura 6: Interacción de los AU y el servidor Proxy SIP	79
Figura 7: Interacción de los AU y el servidor Proxy de Redireccionamiento	80
Figura 8: Ejemplo de Solicitud INVITE	99
Figura 9: Proceso de Autenticación del Mensaje SIP	112
Figura 10: Dificultad que presenta el protocolo SIP al ingresar o salir de la red a través del NATEO	115
Figura 11: Esquema de funcionamiento del servidor STUN	116
Figura 12: Esquema Del Funcionamiento de PIX Para PAT.	117
Figura 13: Dos personas comunicándose a partir del servidor SIP.	121
Figura 14: Establecimiento de la llamada a partir del servidor SIP	121
Figura 15: Mensaje de Respuesta a la petición	122
Figura 16: El envío de un 200 ok, indica que se ha establecido la comunicación	123
Figura 17: Cuando se ha establecido la comunicación el servidor responde con un ACK	124
Figura 18: Petición BYE que es enviada cuando se quiere finalizar la llamada.	125
Figura 19: Se acepta finalización de la llamada con un OK	126
Figura 20: Servidor envía Un INVITE al usuario solicitando la conexión	127
Figura 21: El usuario envía una respuesta temporal	127
Figura 22: Respuesta OK, notifica que se ha establecido la comunicación.	128
Figura 23: El servidor responde con un mensaje ACK	129
Figura 24: Cuando se finaliza la comunicación el servidor transmite la solicitud a través de la petición BYE	130
Figura 25: El mensaje ok confirma que la comunicación ha finalizado	130
Figura 26: Esquema de arquitectura de una PBX	130

Figura 27: Diagrama de bloques que muestra la unidad de conmutación y la de control.	134
Figura 28: Esquema conceptual del funcionamiento de Asterisk	135
Figura 29: Ejemplo de sistema centralizado	142
Figura 30: Ejemplo de sistema distribuido	142
Figura 31: Esquema de la arquitectura de asterisk.	147
Figura 32: Centralita Nueva con conexiones y extensiones de varios tipos	162
Figura 33: Pasarela para dotar a una centralita tradicional de servicios nuevos	163
Figura 34: Pasarela para dotar a una centralita tradicional de nuevas extensiones	163
Figura 35: Varias Oficinas con un Asterisk Centralizado	163
Figura 36: Varias Oficinas con Sistemas Asterisk interconectados	164
Figura 37: Diagrama y elementos de de la Red a implementar.	167
Figura 38: Primera pantalla de arranque de AsteriskNow	168
Figura 39: Pantalla de bienvenida de AsteriskNow	169
Figura 40: Pantalla de selección de tipo de instalación.	169
Figura 41: Pantalla de advertencia sobre la pérdida de particiones.	170
Figura 42: Pantalla de partición automática.	170
Figura 43: Mensaje de Advertencia que se eliminaran las particiones existentes.	170
Figura 44: Configuración de la Red.	171
Figura 45: Configuración de la zona horaria.	171
Figura 46: Pantalla de Ingreso de Password del usuario Admin.	172
Figura 47: Resumen de los procesos que se llevaran a cabo en la instalación.	172
Figura 48: Formateo del sistema	173
Figura 49: Inicio de la Instalación de Aterisk.	173
Figura 50: Finalización de la Instalación de Aterisk.	174
Figura 51: Solicitud de reinicio del equipo.	174
Figura 52: Pantalla de Bienvenida.	175
Figura 53: Pantalla principal de AsteriskNow	175
Figura 54: Asterisk a través del navegador.	175

Figura 55: Pantalla principal de la administración de Asterisk vía Web	176
Figura 56: Configuración de Putty para ingresar vía consola al servidor Asterisk	177
Figura 57: Pantalla de registro de usuario del servidor Asterisk	177
Figura 58: Ingreso al editor nano para editar el archivo sip.conf	177
Figura 59: Configuración del archivo sip.conf	178
Figura 60: Ingreso al archivo extensions.conf	178
Figura 61: Configuración del archivo extensions.conf	179
Figura 62: Monitoreo de las acciones del servidor	179
Figura 63: Verificación del estado de los usuarios.	180
Figura 64: Activación del Debug.	180
Figura 65: Ingreso de la IP del servidor al Teléfono IP.	181
Figura 66: Configuración de extensión y autenticación del teléfono.	181
Figura 67: Configuración de códec.	182
Figura 68: Pantalla de ingreso a la configuración de adaptador	182
Figura 69: Configuración de SIP Configuración.	183
Figura 70: Configuración de Provisioning Server	183
Figura 71: Pantalla de Instalación de X-Lite	184
Figura 72: configuración del Softphone X-Lite	184
Figura 73: configuración de la cuenta de SIP	185
Figura 74: Configuración de los parámetros del Softphone	185
Figura 75: Softphone Muestra la extensión asignada.	186
Figura 76: Establecimiento de la llamada entre Softphone y Teléfono IP.	186

Introducción

El presente documento está formado por cinco capítulos que tiene por objetivo explicar el funcionamiento de la tecnología Voz sobre IP (VoIP); en la que se destaca el uso del protocolo de inicio de sesión SIP, como el encargado de establecer las sesiones de comunicación entre los diferentes usuarios, además se desarrolla un ejemplo práctico de su configuración en la central telefónica Asterisk.

El segundo capítulo detalla las generalidades de lo que es la tecnología VoIP, como lo es su historia, funcionalidades, estructura y aplicaciones; además se explica la diferencia existente y las ventajas que presenta entre el funcionamiento de la telefonía tradicional y la tecnología VoIP.

El tercer capítulo, es centralizado en dar a conocer en detalle uno de los protocolos esenciales en el desarrollo de la tecnología VoIP, refiriéndose este al protocolo de Inicio de Sesión SIP. Donde se explica su estructura, funcionamiento y la interacción que tiene entre los diferentes elementos que integran una red de comunicación VoIP, que trabaja utilizando el protocolo de inicio de sesión.

La central telefónica por software Asterisk es ampliamente conocida en el entorno de VoIP, debido a sus grandes funcionalidades; pero sobre todo por su ventaja de ser código abierto. Las generalidades y el funcionamiento de esta herramienta poderosa son detalladas en el capítulo cuatro.

Por último se implementa los conocimientos adquiridos en los capítulos anteriores, desarrollando un ejemplo de aplicación con un servidor Asterisk que utiliza el protocolo SIP, para permitir el funcionamiento de una red de comunicación VoIP.

CAPITULO I

1.1 Antecedentes

Actualmente se presenta un acelerado incremento en la demanda de aplicaciones que requieren videoconferencias, multimedia y respuestas en tiempo real por parte de usuarios de Internet y redes privadas, lo que ha generado que las tecnologías para comunicarse se mantengan en constante evolución.

Una de las más importantes evoluciones presentadas actualmente es la tecnología Voz sobre IP, que permite que las redes originalmente diseñadas para el transporte de datos, puedan al mismo tiempo ser utilizadas para transmitir comunicación multimedia; para lograr dicha integración de servicios se requiere de mecanismos y protocolos especiales.

Entre los protocolos utilizados para la comunicación de VoIP, se destacan los protocolos de señalización, que se encargan de crear, modificar, y terminar sesiones con uno o más participantes de una sesión multimedia; siendo SIP uno de los protocolos de señalización más utilizados actualmente.

La aplicación de SIP en una red VoIP requiere de elementos encargados de gestionar la comunicación entre los puntos que la integran; en el caso de una red telefónica se necesita una central PBX para realizar dicha función. El software Asterisk permite tener en un mismo dispositivo, la central PBX para los elementos de telefonía tradicional y los elementos que trabajan con tecnología VoIP.

Gracias a este desarrollo de tecnologías de comunicación; día con día surge la necesidad en las personas de conocer más sobre los elementos involucrados en el funcionamiento de dichas tecnologías; es así como nace el interés y la necesidad de facilitar información sobre la tecnología VoIP y a la vez proporcionar guías de implementación que faciliten al usuario su comprensión y uso.

1.2 Importancia De La Investigación

La tecnología IP, ha incrementado considerablemente durante los últimos años; esta es utilizada por operadores de telefonía nacional e internacional, servicios de Internet, hasta pequeñas y aplicación de tecnologías que permiten la comunicación de voz a través de redes grandes empresas.

Esta tecnología de comunicación, hace uso de protocolos de señalización que permiten crear, modificar, y terminar sesiones multimedia con los diferentes usuarios de la red, de todos los protocolos que se han desarrollado para tal función, el Protocolo de Inicio de Sesión SIP, se ha convertido en el mas utilizado en el funcionamiento de aplicaciones, en los servicios de telefonía privada, y por muchos de los diseñadores de dispositivos de voz sobre IP.

El software libre Asterisk es la central telefónica sin costo, más utilizada que permite implementar todas las funciones de una central telefónica IP, cuenta con hardware diseñado específicamente para el funcionamiento de Asterisk, además de permitir su configuración con los distintos protocolos de señalización.

El desarrollo de este proyecto permite dar a conocer los aspectos importantes del protocolo de inicio de sesión SIP; al igual que las características de funcionamiento e implementación del software libre Asterisk, que permita la configuración de una solución de VoIP, que utilice SIP como protocolo de señalización.

1.3 Planteamiento del Problema.

La tecnología VoIP creada para transportar comunicación telefónica a través de redes IP, ha evolucionado en los últimos años, debido entre otras cosas, al mayor aprovechamiento del ancho de banda y a su menor costo comparada con la telefonía tradicional.

Aplicar esta nueva tecnología implica el uso de uno de los protocolos de señalización, entre los cuales sobresale el protocolo de inicio de sesión SIP, que durante los últimos años se ha convertido en el más utilizado para el desarrollo de aplicaciones en Internet y en redes de voz. Por lo que se necesita saber los detalles de su funcionamiento, las ventajas que posee sobre los demás protocolos diseñados, los elementos necesarios para desarrollar una red SIP, y al mismo tiempo, es importante conocer el funcionamiento, el proceso de instalación y la configuración de un Software PBX, que permita la comunicación entre los dispositivos de la red. De esta manera se podrá implementar una red SIP perfectamente funcional para la comunicación de voz en un ambiente privado.

1.4 Definición del Tema

"Estudio monográfico del protocolo SIP y su aplicación en servicios de voz sobre IP."

Descripción General del Tema

El tema inicia con una investigación sobre los protocolos encargados del establecimiento de la comunicación, como es el caso del protocolo de inicio de sesión, SIP. Al mismo tiempo detallara el funcionamiento de los dispositivos que integran las redes SIP y sus descripciones.

La investigación detallara aspectos importantes de SIP, tales como: historia, funcionamiento, rol que toma en la realización de la comunicación telefónica, problemas que presenta actualmente y posibles soluciones. Por otro lado cubre las características del funcionamiento y configuración del software libre de central telefónica (PBX) por Voz sobre IP Asterisk. Además se implementara una red de comunicación VoIP utilizando SIP a través de Asterisk, para demostrar las ventajas de las redes IP, permitiendo la interacción entre: computadoras, teléfonos análogos, teléfonos IP y comunicación remota a través de redes virtuales.

1.5 Justificación

La transición que eventualmente darán las comunicaciones telefónicas y multimedia a tecnología VoIP, hace necesario que las personas involucradas en la implementación de esta tecnología; posean los conocimientos sobre aspectos tan importantes como los protocolos de señalización, especialmente conocer el funcionamiento y configuración del Protocolo de Inicio de Sesión SIP, ya que este apunta a convertirse en un estándar internacional por su rápida aceptación por parte de los desarrolladores de tecnología VoIP.

El desarrollo y aplicación de SIP en servicios de VoIP esta ampliamente difundido internacionalmente, en El Salvador por el contrario, a nivel de educación superior es poca o nula la información proporcionada sobre esta tecnología; por lo que este proyecto pretende servir en un futuro como herramienta didáctica para comprender e implementar un red de VoIP al mismo tiempo como referencia a futuras investigaciones y desarrollo de aplicaciones que involucren la utilización del protocolo SIP.

El conocer los aspectos principales del funcionamiento y configuración del software libre Asterisk permite a los profesionales construir y configurar una central telefónica PBX, perfectamente funcional y con las mismas características de equipos de comunicación utilizados en redes de voz sobre IP y redes tradicionales, que son adquiridos a costos elevados.

1.6 Objetivos

1.6.1 Objetivo General.

- ✓ Investigación del funcionamiento y características del protocolo SIP y desarrollo de su aplicación sobre una solución de VoIP.

1.6.2 Objetivos Específicos

- Investigar de forma comparativa los protocolos para el despliegue de servicios de VoIP.
- Describir el funcionamiento del protocolo SIP y sus aplicaciones.
- Describir el funcionamiento y aplicaciones del software libre de central telefónica (PBX), Asterisk.
- Implementar una solución de VoIP utilizando un servidor Asterisk.
- Determinar los requerimientos necesarios para el adecuado funcionamiento del servidor Asterisk, para el despliegue de VoIP.
- Generar una guía de implementación de un servidor Asterisk como una solución de voz sobre IP

1.7 Alcances

- Realizar una investigación que describa los protocolos encargados de la comunicación de Voz en una red IP, haciendo énfasis en el Protocolo de Inicio de Sesión SIP.
- Elaborar una referencia técnica que permita la implementación de un servidor Asterisk que funcione bajo protocolo SIP y permita la comunicación de voz sobre IP.
- Realizar una aplicación sobre el funcionamiento del protocolo SIP, donde se establezca la comunicación telefónica utilizando: terminales, teléfonos IP, teléfonos análogos (a través de los adaptadores telefónicos análogos) y un servidor configurado con Asterisk.
- Permitir la comunicación telefónica entre Computadoras a través de Teléfonos IP y teléfonos análogos utilizando adaptadores (ATA), conectados en una misma red.

1.8 Limitaciones

- El estudio comparativo se limitara a los siguientes protocolos de estándar abierto, tales como: H323, SIP e IAX2.
- La investigación de aplicaciones del protocolo SIP se hará únicamente para servicios de VoIP.
- Para lograr la comunicación VoIP entre el servidor Asterisk con teléfonos análogos se hará uso de adaptadores ATA y no de tarjetas Digium debido a su elevado costo para fines demostrativos.
- La configuración del servidor Asterisk, únicamente será desarrollado bajo el sistemas operativo, Linux.
- La solución se implementara en un servidor experimental, por lo que el numero de usuarios estará limitado a pocos teléfonos IP
- Las técnicas de QoS no se detallaran durante este estudio monográfico, ya que existen trabajos anteriores sobre el tema.

1.9 Delimitación.

El estudio se basa en una investigación sobre el protocolo SIP y sus aplicaciones en la tecnología VoIP, el desarrollo del proyecto se cubrirá en un periodo de ocho meses, dando inicio el 24 de marzo del 2008 y finalizando el 16 de septiembre del mismo año. La investigación cubrirá todos los puntos necesarios para llevar a cabo una muestra del funcionamiento del protocolo a través de la implementación de una red que conectara dos puntos; dicha red se realizara en las instalaciones del laboratorio de cisco de la universidad Don Bosco en un periodo que cubre del 23 de junio del 2008 al 03 de agosto 2008.

1.10 Marco Teórico

1.10.1 Referencias Históricas

SIP o Protocolo de Inicio de Sesiones es un protocolo desarrollado por el IETF MMUSIC Working Group con la intención de ser el estándar para la iniciación, modificación y finalización de sesiones interactivas de usuario donde intervienen elementos multimedia como el video, voz, mensajería instantánea, juegos online y realidad virtual.

Su aparición surge el 22 de febrero de 1996, cuando Mark Handley y Eve Schooler presentaron al IETF un borrador del Protocolo de invitación de sesión, conocido ahora como SIPv1. El mismo estaba basado en trabajos anteriores de Thierry Turlitti (INRIA sistema de videoconferencia o IVS) y de Eve Schooler (Control de Conferencia Multimedia o MMCC). Su principal fortaleza, heredada por la versión actual de SIP, era el concepto de registración, por el cual un usuario informaba a la red dónde (en qué host de Internet) podía recibir invitaciones a conferencias. Ese mismo día el Dr. Henning Schulzrinne presentó un borrador del Protocolo de Invitación de Conferencia (SCIP), que estaba basado en el HTTP (*Protocolo de Transporte de Hipertexto*). Usaba TCP (Protocolo de control de transmisión) como protocolo de transporte. Como identificadores de los usuarios utilizaba direcciones de correo electrónico para permitir el uso de una misma dirección para recibir correos electrónicos e invitaciones a conferencias multimedia. No utilizaba al SDP¹ para la descripción de los contenidos sino que creaba un mecanismo propio.

El IETF decidió combinar ambos en un único protocolo denominado Protocolo de Inicio de Sesión, (es decir cambiando el significado de la inicial en el acrónimo "SIP") y su número de versión fue el dos, dando origen al SIPv2. En diciembre de 1996 los tres autores (Schulzrinne, Handley y Schooler), presentaron el borrador del SIPv2. El mismo luego de ser discutido en el grupo de trabajo MMUSIC (Control de Sesión

¹ Protocolo de Descripción de Sesión (SDP), es un formato para la descripción de medios de los parámetros de inicialización. It has been published by the IETF as RFC 4566. Se ha publicado por la IETF como RFC 4566.

Multimedia Partida) del IETF alcanzó el grado de "Propuesta Estándar" en la RFC 2543 publicada en Febrero de 1999. En septiembre de 1999 se creó el grupo de trabajo SIP en el IETF que continuó con el desarrollo del protocolo y en noviembre del año 2000, SIP fue aceptado como el protocolo de señalización de 3GPP y elemento permanente de la arquitectura IMS (*IP Multimedia Subsystem*). En Junio de 2002 se publicó la RFC 3261 que reemplazó a la anterior introduciendo modificaciones propuestas durante el trabajo del grupo SIP. Los autores de esta última RFC, hoy vigente son: Jonnathan Rosenberg, Henning Schulzrinne, Gonzalo Camarillo, Allan Johnston, Jon Peterson, Robert Sparks, Mark Handley y Eve Schooler.

Asterisk es la aplicación de centralita telefónica PBX bajo licencia GPL, fue desarrollada por Mark Spencer, entonces estudiante de ingeniería informática en la Universidad de Auburn, Alabama. Mark había creado en 1999 la empresa "Linux Support Services" con el objetivo de dar soporte a usuarios de Linux. Para ello necesitaba una centralita telefónica, pero ante la imposibilidad de adquirirla dados sus elevados precios, decidió construir una con un PC bajo Linux, utilizando lenguaje C.

Pero Actualmente Asterisk también funciona en BSD, MacOSX, Solaris y Microsoft Windows, aunque la plataforma nativa (GNU/Linux) es la mejor soportada de todas.

El 15 de Noviembre del 2005 se publica el establecimiento de la versión Asterisk 1.2; posteriormente la última versión que actualmente se utiliza es Asterisk 1.2.9.1 que se libero el 6 de junio del 2006. Dicha versión esta compuesto por los módulos siguientes:

- Asterisk: Ficheros base del proyecto.
- Zaptel: Soporte para hardware. Drivers de tarjetas.
- Addons: Complementos y añadidos del paquete Asterisk. Opcional.
- Libpri: Soporte para conexiones digitales. Opcional.
- Sounds: Aporta sonidos y frases en diferentes idiomas.

Donde cada módulo cuenta con una versión estable y una versión de desarrollo.

1.10.2 Marco Conceptual.

➤ Red de Telefonía Pública.

La operación de la red telefónica inicia con la conexión directa entre cada uno de los elementos integraban la red telefónica, debido a las limitaciones de escalabilidad y administración, surgen las centrales telefónicas, que funcionaban como puntos centrales de red encargadas de interconectar las líneas entre los elementos que deseaban comunicarse.

La interconexión de cada una de estas centrales telefónicas es lo que se conoce como “Red Publica de Conmutación Telefónica” (PSTN). Esta red funciona a través de múltiples niveles de switches, que interconectan a los usuarios más próximos, si no están conectado al mismo dispositivo, sube al siguiente nivel y así de manera sucesiva hasta ubicar al destinatario de la conexión.

Elementos de Red Telefónica

La red de telefonía consta de los siguientes elementos.

- El Aparato Telefónico o Teléfono: es el aparato con el cual nos conectamos a la red
- El Bucle Local (Local Loop): La conexión física entre la compañía telefónica local y nuestro teléfono se realiza a través de dos hilos de cobre, denominados comúnmente, cada uno de ellos, como **Tip y Ring**. Este par de hilos conforman el **Bucle Local (Local Loop)** y son el medio físico para la ruta de señalización y de comunicación entre la compañía telefónica y nuestra casa, a la que se le denomina línea telefónica.
- La Línea Troncal: son el medio de conexión entre Centrales. Estos enlaces o líneas tróncales pueden ser una simple línea de dos hilos que conecta una

pequeña PBX con la Oficina Central o un enlace digital T1/E1 entre Centrales Públicas

- El Conmutador o Switch de Voz (Central Pública o Privada): identifica, direcciona y asegura una ruta entre dos usuarios para establecer una llamada. De acuerdo a la función que desempeñe el Switch este se puede clasificar en tres tipos:
 - Switch de Oficina Central (Central Office Switch).
 - Switch Tándem o Intermedio (Tandem Switch).
 - PBX (Private Branch Exchange).

Los dos primeros tipos de Switch se conocen mejor por el nombre de **Centrales Públicas** y la PBX como **Central Privada**.

Redes de Conmutación de Circuitos

La comunicación entre dos estaciones utilizando conmutación de circuitos implica la existencia de un camino dedicado entre ambas estaciones. Dicho camino está constituido por una serie de enlaces entre algunos de los nodos que conforman la red. En cada enlace físico entre nodos, se utiliza un canal lógico para cada conexión. Esto se denomina circuitos virtuales y en un escenario ideal los usuarios del circuito no perciben ninguna diferencia con respecto a un circuito físico y no tienen conocimiento del uso compartido de circuitos físicos.

Una comunicación mediante circuitos conmutados posee tres etapas bien definidas

Establecimiento del Circuito

Cuando un usuario quiere obtener servicios de red para establecer una comunicación se deberá establecer un circuito entre la estación de origen y la de destino. En esta etapa dependiendo de la tecnología utilizada se pueden establecer la capacidad del canal y el tipo de servicio.

Transferencia de datos

Una vez que se ha establecido un circuito puede comenzar la transmisión de información. Dependiendo del tipo de redes y del tipo de servicio la transmisión será digital o analógica y el sentido de la misma será unidireccional o full duplex.

Cierre del circuito

Una vez que se ha transmitido todos los datos, una de las estaciones comienza la terminación de la sesión y la desconexión del circuito. Una vez liberado los recursos utilizados por el circuito pueden ser usados por otra comunicación.

En una conmutación por circuitos, la capacidad del canal se reserva al establecer el circuito y se mantiene durante el tiempo que dure la conexión, incluso si no se transmiten datos.

Proceso de Señalización.

En las redes de telecomunicaciones la señalización es de importancia fundamental para establecimiento de la llamada o enlace de datos, conexión, desconexión, y facturación.

La señalización entonces, es el conjunto de comandos y configuraciones que permitirán el establecimiento de enlaces de comunicaciones en voz, data fax y aún video en dos, tres o más abonados o hosts, según sea el caso.

➤ Tecnología VoIP.

El término Voz sobre IP, VoIP por sus siglas en inglés (Voice Over Internet Protocol), se refiere a la utilización de redes IP y sus protocolos para comunicaciones de voz, permitiendo que la voz al igual que los datos utilice el mismo medio de transmisión.

Se distinguen dos diferencias para la aplicación de VoIP en servicios de telefonía. El primero es cuando la VoIP es transportada a través de redes privadas o

empresariales, y el segundo cuando el internet es utilizado para transportar la comunicación de dos extremos. En este caso se conoce como Telefonía por Internet. La diferencia no sólo solo es el medio de transporte sino también en las posibilidades de establecer mecanismos de control (QoS) que garanticen la máxima calidad de servicio en todo momento. Los mecanismos y técnicas aplicadas en ambos casos difieren sustancialmente, pero a través de éstas se consiguen niveles de calidad de servicio similares a la telefonía convencional PSTN.

La voz sobre IP tiene dos modos de ser transportado:

- A través de líneas privadas y dedicadas que proporcionan una calidad de servicio aceptable.
- A través de redes publicas como Internet o redes publicas IP con una calidad de servicio inferior

Los parámetros más influyentes a la hora de transmitir la voz sobre IP son:

A. Retardo

El retardo causa dos problemas: eco y traslape del habla. El eco es causado por las señales reflejadas por el equipo telefónico del extremo distante que regresan al oído del hablante.

El eco llega a ser un problema significativo cuando el retardo del viaje redondo llega a ser más de 50 milisegundos. A medida que el eco se incrementa, los sistemas de paquetes se ven en la necesidad de utilizar controles como la cancelación de eco.

B. Jitter

Es la variación de tiempo entre los paquetes causada por la red. Remover el jitter requiere a recolección de paquetes y retención de estos el tiempo suficiente para que el paquete más lento llegue a tiempo para ser interpretado en la secuencia correcta.

C. Compensación de Perdida de Paquetes

La perdida de paquetes puede ser un problema aun mayor dependiendo del tipo de

red de paquetes que este siendo usada. Ya que la red IP no garantiza el servicio, usualmente tiene mayor perdida de paquetes que las redes ATM.

D. Compensación de Eco

El eco en una red telefónica es causado por las reflexiones de señales generadas por un circuito hibrido que convierte de 4 hilos (un par para transmisión y uno para recepción) a 2 hilos (un solo hilo para transmisión y uno para recepción). Estas reflexiones de la voz del hablante son escuchadas por el oyente. El eco se presenta aun en las redes de conmutación de circuitos, sin embargo es aceptable ya que los retardos completos a través de la red son menores que 50 msg y el eco es enmascarado por el tono lateral que todo teléfono genera.

El eco es problema en una red de paquetes de voz cuando el retardo completo en la red es mayor que 50 msg, entonces se deben aplicar técnicas de cancelación de eco.

Telefonía Por Internet

Telefonía por Internet (Internet Telephony) se refiere a VoIP sobre Internet.

Aunque es técnicamente posible la calidad de la llamada es muy variable debido a la falta de sistemas de calidad de servicio QoS sobre Internet.

Calidad de Servicios QoS.

Su termino se refiere a la capacidad de una red de proporcionar un conjunto de características que adaptan la entrega de tráfico a los requisitos del usuario, es decir, QoS hace posible que los administradores de la red pueden tener un mejor control sobre sus redes, reducir los costos, y mejorar la satisfacción de cliente.

Al implementar QoS se logra mejorar, haciendo más confiable el servicio, las siguientes capacidades de la red:

- Soporte de ancho de banda dedicado

- Mejora en la cantidad de paquetes perdidos
- Administración y prevención de congestiones en la red
- Configuración de tráfico en la red
- Prioridades al tráfico, para usos a través de la red.

Conmutación Paquetes

La tecnología VoIP significa que la señal de voz se envía en forma de paquetes de datos en lugar de enviarla a través de circuitos conmutados utilizando la PSTN.

En conmutación de paquetes, los datos se transmiten en paquetes cortos. Para transmitir grupos de datos más grandes, el emisor trocea estos grupos en paquetes más pequeños y les adiciona una serie de bits de control. En cada nodo, el paquete se recibe, se almacena durante un cierto tiempo y se transmite hacia el emisor o hacia un nodo intermedio.

Hay dos técnicas básicas para el envío de estos paquetes:

1. Técnica de datagramas: cada paquete se trata de forma independiente, es decir, el emisor enumera cada paquete, le añade información de control (por ejemplo número de paquete, nombre, dirección de destino, etc.) y lo envía hacia su destino. Puede ocurrir que por haber tomado caminos diferentes, un paquete con número por ejemplo 6 llegue a su destino antes que el número 5. También puede ocurrir que se pierda el paquete número 4. Todo esto no lo sabe ni puede controlar el emisor, por lo que tiene que ser el receptor el encargado de ordenar los paquetes y saber los que se han perdido (para su posible reclamación al emisor), y para esto, debe tener el software necesario.
2. Técnica de circuitos virtuales: antes de enviar los paquetes de datos, el emisor envía un paquete de control que es de Petición de Llamada, este paquete se encarga de establecer un camino lógico de nodo en nodo por donde irán uno a uno todos los paquetes de datos. De esta forma se establece un camino virtual para todo el grupo de paquetes. Este camino virtual será

numerado o nombrado inicialmente en el emisor y será el paquete inicial de Petición de Llamada el encargado de ir informando a cada uno de los nodos por los que pase de que más adelante irán llegando los paquetes de datos con ese nombre o número. De esta forma, el encaminamiento sólo se hace una vez (para la Petición de Llamada). El sistema es similar a la conmutación de circuitos, pero se permite a cada nodo mantener multitud de circuitos virtuales a la vez.

Las redes de datos, basadas en la conmutación de paquetes, se identifican por las siguientes características:

- Para asegurar la entrega de los datos se requiere el direccionamiento por paquetes, sin que sea necesario el establecimiento de llamada.
- El consumo de los recursos de red se realiza en función de las necesidades, sin que, por lo general, sean reservados siguiendo un criterio de extremo a extremo.
- Los precios se forman exclusivamente en función de la tensión competitiva de la oferta y la demanda.
- Los servicios se prestan de acuerdo a los criterios impuestos por la demanda, variando ampliamente en cuanto a cobertura geográfica, velocidad de la tecnología aplicada y condiciones de prestación.

Protocolos de Señalización VoIP.

Al igual como se describió en las redes de voz, los mecanismos de señalización y control de llamadas, en las redes de paquetes, también existen estos mecanismos. Los protocolos que realizan el trabajo de establecer, administrar, controlar y terminar una llamada en una red de paquetes son conocidos como protocolos de señalización o procesamiento de llamada.

Se han desarrollado diferentes soluciones para la señalización de control de llamada en sistemas de VoIP, que son los siguientes:

- Modelo H.323
- Modelo SIP

- Modelo IAX

Estos modelos son soluciones diferentes a la misma problemática, la señalización de control de llamada para el servicio de VoIP, cada una con una arquitectura funcional y protocolos que la caracterizan.

Codec

Para lograr que la voz viaje sobre Internet esta tiene que ser digitalizada y comprimida, a partir de un proceso que convierte ondas analógicas a información digital con un codificador-decodificador llamado CODEC.

Existen codecs tanto de audio como de vídeo según se quiera transmitir sonido o imagen.

Central Telefónica Privada (PBX)

Una *Central secundaria privada automática PBX (Private Branch Exchange)*, es cualquier central telefónica conectada directamente a la red pública de teléfono por medio de líneas troncales para gestionar, además de las llamadas internas, las entrantes y/o salientes con autonomía sobre cualquier otra central telefónica. Este dispositivo generalmente pertenece a la empresa que lo tiene instalado y no a la compañía telefónica.

Un PBX se refiere al dispositivo que actúa como un ramificación de la red primaria pública de teléfono, por lo que los usuarios no se comunican al exterior mediante líneas telefónicas convencionales, sino que al estar el PBX directamente conectado a la PSTN, será esta misma la que enrute la llamada hasta su destino final mediante enlaces unificados de transporte de voz llamados líneas troncales. Es decir, los usuarios de una PBX no tienen asociada ninguna central de teléfono pública, ya que es el mismo PBX que actúa como tal, Así como una central pública da cobertura a todo un sector una central PBX lo ofrece a las instalaciones de una compañía.

Un PBX mantiene tres funciones esenciales:

- Establecer llamadas entre dos o más usuarios. (Llamadas internas o externas)
- Mantener la comunicación durante el tiempo que lo requiera el usuario.
- Proveer información para contabilidad y/o facturación de llamadas.

Funcionamiento

Los PBX de gran escala, instalados en grandes oficinas funcionan como dispositivo físico que administra el tráfico de llamadas. Generalmente el usuario conecta el PBX por un único enlace digital, como E1 ó T1, utilizando tan solo 2 pares de cables en lugar de $2n$ hilos para las n líneas externas contratadas. Generalmente estos enlaces tienen capacidad de portar hasta 30 líneas sin llegar a comprimir la información de la voz lo suficiente como para degradarla.

➤ **Asterisk**

Asterisk es una centralita software (PBX) de código abierto, lo que significa que cientos de desarrolladores están trabajando todos los días en el.

Como cualquier centralita PBX permite interconectar teléfonos y conectar dichos teléfonos a la red telefónica convencional (RTB - Red telefónica básica)- Su nombre viene del símbolo asterisco (*) en ingles.

El creador original de esta centralita es Mark Spencer de la compañía Digium que sigue siendo el principal desarrollador de las versiones estables. Pero al ser de código libre, existen multitud de desarrolladores que han aportado funciones y nuevas aplicaciones. Originalmente fue creada para sistemas Linux pero hoy en día funciona también en sistemas OpenBSD, FreeBSD, Mac OS X, Solaris Sun y Windows. Pero Linux sigue siendo la que mas soporte presenta.

1.10.3 Marco Experimental.

Existen investigaciones previas relacionadas con la tecnología VoIP en la Universidad Don Bosco; entre ellas encontramos:

“Estudio Técnico para la implementación de una solución de voz sobre IP en una Red de Datos Corporativa”.

Este documento indica los puntos a tomar en cuenta para la implementación de voz sobre IP en redes corporativas. Proporciona la especificación de protocolos para comprender como se gestiona una llamada, y brinda un ejemplo detallado de gestión de llamada para uno de los protocolos más utilizados.

“Monografía acerca de la transmisión de voz sobre los protocolos Frame Relay e IP”

Brinda una referencia teórica, para estudiantes, profesores y profesionales de Ingeniería en el área de las Telecomunicaciones y Redes de Datos, para entender como funcionan las tecnologías de voz sobre redes de datos, que están siendo aplicadas en el ámbito tecnológico del país. El desarrollo del documento aborda específicamente los temas de Voz sobre Frame Relay y Voz sobre IP.

“Integración de Voz, Video y Datos sobre Frame Relay, ATM e IP”

Que trata sobre las tecnologías WAN que puedan soportar voz, video y datos; Esta abarca el estudio de los dispositivos, empaquetados, estándares de señalización y protocolos que se encargan de realizar el transporte de voz, video y datos sobre la red de paquetes.

1.11 Metodología de La Investigación

El estudio que se llevara a acabo sobre el protocolo SIP y su aplicación, requiere de la investigación de varios elementos que participan en el desarrollo e implementación de este protocolo, sobre los servicios de VoIP.

Es así como se ha definido las siguientes faces de trabajo a realizar, para llevar acabo el proyecto en su plenitud.

Recopilación de La Información

Esta fase comprende la recopilación de toda aquella información necesaria y requerida para llevar acabo la investigación y cumplir con las expectativas trazadas al inicio del proyecto. La información será obtenida a partir de:

- **Investigación Bibliográfica.**

Se realizara la investigación para obtener las características principales del protocolo SIP, y del funcionamiento del software Asterisk a través de: libros, libros en medios electrónicos, artículos publicados en revistas especializadas, y trabajos de graduación realizados con anterioridad.

- **Información Disponible En Internet.**

Esta fuente de información nos permitirá recopilar datos a través de Sitios Web, documentos electrónicos, artículos publicados en otras universidades o directamente desde los sitios involucrados en el desarrollo de la tecnología VoIP, específicamente información de la aplicación de SIP, al mismo tiempo ejemplos de configuración del Software Asterisk.

- **Asesoría Profesional.**

Realizar sesiones con profesionales que estén involucrados en la utilización e implementación del protocolo SIP en redes de VoIP, al mismo tiempo consultar sobre el desenvolvimiento y el proceso de configuración del software Asterisk. Así

obtendremos información práctica del desenvolvimiento de estas tecnologías en un entorno profesional.

Esta etapa de recopilación de información es de suma importancia para las fases posteriores, ya que establece el camino por el cual se debe continuar la investigación.

Documentación y Análisis.

Este apartado tiene como función principal dar la estructura al documento, a partir de la información compilada con anterioridad. Consistirá en clasificar los aspectos importantes del estudio que permitan la redacción del documento final, este punto requiere de la máxima comprensión del material investigado en la fase anterior por parte de los autores, debido a que su contenido es lo que se expondrá ante los lectores

Implementación.

Esta fase es una de las más importantes, consiste en poner en práctica toda la investigación realizada con anterioridad. En este punto se llevara a cabo la implementación de una red, que poseerá los elementos mínimos necesarios para llevar a cabo la demostración de una central telefónica, con la tecnología VoIP que utilice el protocolo SIP y el software Asterisk

1.12 Plan de Acción y Cronograma

Inicio del trabajo de graduación lunes 24 de marzo 2008, Finalización del trabajo de graduación viernes 12 de septiembre 2008.

1.12.1 Plan De Acción 1DIA = 3HORAS/HOMBRES

TEMA Macroactividades	ACTIVIDAD Microactividades	OBJETIVO	TIEMPO DE INVESTIGACIÓN	TIEMPO DE REDACCIÓN	RESPONSABLE	VERIFICACIÓN
2. Introducción a la tecnología voip 2.1 Generalidades de la Telefonía 2.1.1 Evolución histórica de la telefonía. 2.1.2 Funcionamiento de la telefonía Actual. 2.1.3 Características de la telefonía. 2.2 Descripción de la telefonía IP 2.2.1 Historia y generalidades de las tecnologías de VoIP 2.2.2 Elementos de integran un sistema de VoIP	Efectuar una breve investigación y resumen sobre el funcionamiento de la telefonía analógica, utilizando: libros, revistas especializadas e internet.	Distinguir la diferencia entre la tecnología Voip y la telefonía Analógica	8 hrs/hombre (24/03/08-30/03/08)	7 hrs/hombre (24/03/08-30/03/08)	Sandra Hdez	Describir los aspectos mas importantes de la telefonía VoIP
	Realizar una investigación sobre el funcionamiento y los elementos que permiten el desarrollo de la tecnología voz sobre IP, utilizando: libros, revistas especializadas e internet.	Describir los elementos que permiten el desarrollo de una solución de VoIP	8 hrs/hombre (24/03/08-30/03/08)	7 hrs/hombre (24/03/08-30/03/08)	Rodrigo Castro	Diagrama de la arquitectura de la tecnología VoIP
2.2.3 Funcionamiento de los sistemas de VoIP 2.2.3.1 Proceso de Codificación de Voz 2.2.3.2 Códec utilizado en VoIP o G.711 o G729A o G723 2.2.4 Protocolos de soporte de VoIP 2.2.4.1 Protocolos de Señalización VoIP. o H323	Realizar una investigación sobre los codecs involucrados en el funcionamiento de la comunicación VoIP:	Detallar las características principales de los protocolos y codecs que permiten la	9 hrs/hombre (31/04/08-06/04/08)	6 hrs/hombre (31/04/08-06/04/08)		Describir la labor que desempeñan los codecs durante la comunicación de VoIP
	□ G.711		3hrs/hombre	2 hrs/hombre	Rodrigo Castro	
	□ G729		3 hrs/hombre	2 hrs/hombre	Sandra Hdez	
	□ G723		3hrs/hombre	2 hrs/hombre	Rodrigo Castro	
	a través libros, revistas especializadas e internet.					
□ H323	3 hrs/hombre	2 hrs/hombre	Sandra Hdez			
□ SIP	3 hrs/hombre	2 hrs/hombre	Rodrigo Castro			

o SIP o IAX2 2.2.5 Características y limitaciones de la VoIP	<input type="checkbox"/> Iax2 por medio de libros, revistas especializadas, internet y personas involucradas con el tema.	comunicacion voz ip	3 hrs/hombre	2 hrs/hombre	Sandra Hdez	
REVISIÓN DEL CAPITULO II " TECNOLOGIA VOIP" POR PARTE DEL ASESOR DEL 07/04/08 AL 13/04/08						
CORRECCION DE OBSERVACIONES REALIZADAS POR PARTE DEL ASCESOR 14/04/08 AL 17/04/08						
ENTREGA DEL CAPITULO I AL LECTOR EL VIERNES 18 DE ABRIL						
REUNION CON EL ASCESOR PARA DEFINIR INDEX CAP III 18/04/08 AL 24/04/08						
3. Protocolo SIP 3.1Historia y desarrollo de estándar SIP 3.2Estructura 3.3Funcionamiento del protocolo SIP 3.4Problema del NAT 3.5Aplicaciones 3.6Funcionamiento de los dispositivos que integran la redes SIP y sus descripciones 3.6.1Agente de usuario SIP 3.6.2Servidor SIP proxy SIP servidor de re-direccionamiento SIP 3.6.3Registro SIP.	Realizar una investigación monográfica sobre las generalidades del protocolo SIP que incluya:	Explicar en detalle los aspectos principales del protocolo SIP, que incluya la historia, estructura, funcionamiento, problemas y aplicaciones.	10 hrs/hombre (25/04/08 AL 04/05/08)	6 hrs/hombre (25/04/08 AL 04/05/08)		Resumen detallado de cada una de los aspectos importantes de SIP.
	Historia		5 hr/hombre	3 hr/hombre	Rodrigo Castro	
	Estructura		5 hrs/hombre	3 hr/hombre	Sandra Hdez, Rodrigo Castro	
	Funcionamiento		5 hrs/hombre	3 hr/hombre		
	Aplicaciones		5 hrs/hombre	3 hr/hombre	Sandra Hdez	
Por medio de libros, revistas especializadas, internet y personas involucradas con el						
	Realizar una investigación y resumir de manera específica el funcionamiento de los elementos y agentes que permiten el existo de una red SIP.	Detalla el funcionamiento de cada uno de los elementos que conforman una red SIP.	10 hrs/hombre (05/05/08-11/06/08)	5 hrs/hombre (05/05/08-11/05/08)	Sandra Hernández, Rodrigo Castro	Nombrar y resumir las características y funciones de los elementos que conforman una red SIP
REVISIÓN DEL CAPITULO III. " PROTOCOLO SIP". POR PARTE DEL ASESOR DEL 12/05/08 AL 18/05/08						
CORRECCION DE OBSERVACIONES REALIZADAS POR PARTE DEL ASCESOR DEL 19/05/08 AL 22/05/08						
ENTREGA DEL CAPITULO II AL LECTOR EL VIERNES 23/05/08						
REUNION CON EL ASCESOR PARA DEFINIR INDEX CAP IV DEL 13/06/08 AL 16/06/08						

4. Solución de VoIP Utilizando Asterisk 4.1 Historia. 4.2 Estructura 4.3 Capacidades 4.4 Limitaciones de la arquitectura de Asterisk 4.5 Aplicaciones de Asterisk	Realizar una investigación monográfica sobre las generalidades Asterisk que incluya:	Explicar en detalle los aspectos principales del Asterisk, que incluya	10 hrs/hombre (30/05/08 -15/05/08)	5 hrs/hombre (30/05/08 - 15/05/08)		Resumen detallado de los aspectos importantes del software libre Asterisk
	Historia	la historia,	3 hrs/hombre	2 hrs/hombre	Sandra Hdez	
	Estructura	estructura,	5 hrs/hombre	3 hrs/hombre	Rodrigo Castro	
	Capacidades	funcionamiento,	3 hrs/hombre	2 hrs/hombre	Sandra Hdez	
	Limitaciones de la arquitectura	problemas y aplicaciones.	5 hrs/hombre	2hrs/hombre	Rodrigo Castro	
Aplicaciones de Asterisk		4 hrs/hombre	2 hrs/hombre	Sandra Hdez,		
REVISIÓN DEL CAPITULO IV. "SOLUCIÓN DE VOIP UTILIZANDO ASTERISK" POR PARTE DEL ASESOR DEL 16/06/08 AL 22/06/08						
CORRECCION DE OBSERVACIONES REALIZADAS PARTE DEL ASCESOR 23/06/08 AL 26/06/08						
ENTREGA DEL CAPITULO III AL LECTOR EL VIERNES 27/06/08						
REUNION CON EL ASCESOR PARA DEFINIR INDEX CAP V DEL 27/06/08 AL 03/07/08						
5. Implementación del Servicio de VoIP 5. Implementación del Servicio de VoIP 5.1 Guía y Configuración del servidor Asterisk para	Siguiendo información recopilada con anterioridad sobre asterisk y SIP se redactara la documentación que detalle los elementos necesarios y requerimientos mínimos de cada dispositivos, para iniciar la configuración del servidor	Generar la información previa necesaria, para lograr llevar acabo la configuración de la red SIP en un entorno Linux	9 hrs/hombre (04 /07/08-10/07/08)	6 hrs/hombre (04 /07/08- 10/07/08)	Sandra Hernández	Lista y documentación detallada de cada proceso que debe realizarse antes de iniciar la configuración del servidor tales: requisitos de hardware y software, requerimientos mínimos entre otros
		Proporcionar una guía de desarrollo que facilite la implementación de la comunicación voip a partir del servidor Asterisk y protocolo SIP			Rodrigo Castro	

establecer una comunicación telefónica en una misma LAN entre dispositivos analógicos y digitales. 5.1.1 Consideraciones Previa 5.1.2 Configuración del servidor 5.1.3 Realización de Pruebas 5.1.4 Depuración de errores. 5.1.5 Buenas practicas para el funcionamiento de servicios de Voz sobre IP	Teniendo en cuenta la actividad anterior se realizara la configuración del software Asterisk bajo el protocolo SIP, en un servidor Linux.	Obtener un servidor en Asterisk configurado con el protocolo SIP, que trabaje de manera optima	12 hrs/hombre (11/07/08-24/07/08)	3 hrs/hombre (11/07/08-24/07/08)	Sandra Hernández, Rodrigo Castro	Un servidor que utilice el software Asterisk para la comunicación de VoIP en una red SIP
	Realizar pruebas de comunicación del servidor Asterisk, entre los diferentes elementos de la red como son: Computadoras, teléfonos IP, teléfonos analógicos	Verificar el funcionamiento del servidor Asterisk, realizando pruebas de comunicación entre PCs, teléfonos IP y teléfonos analógicos, que nos permitan identificar posibles problemas que se pueda presentar.	15 hrs/hombre (25/07/08-03/07/08)	3 hrs/hombre (25/07/08-03/07/08)	Sandra Hernández, Rodrigo Castro	Realizacion de llamadas a través del servidor Asterisk entre pc's por medio softphone, teléfonos Ip y teléfonos analógicos.
	Con las pruebas realizadas y corrección de errores que puedan surgir, se dará una lista de buenas practicas de construcción y configuración de una red VoIP, que trabaje utilizando el protocolo SIP	Proporcionar una guia de buenas practicas para el funcionamiento ideal de una red de voz sobre IP a través del protocolo SIP	3 hrs/hombre (11/08/08-17/08/08)	12 hrs/hombre (11/08/08-17/08/08)	Sandra Hernández, Rodrigo Castro	Guia de buenas practicas para la configuración de software y uso de hardware apropiado, para alcanzar el optimo funcionamiento de la red VoIP a travez de SIP.
REDACCIÓN DEL DOCUMENTO FINAL 18/08/08 AL 21/08/08						
REVISIÓN DEL CAPITULO V. "IMPLEMENTACIÓN DEL SERVIDOR VOIP" POR PARTE DEL ASESOR DEL 22/08/08 AL 28/08/08						
CORRECCION DEL TRABAJO DE GRADUACION DEL 29/08/08 AL 02/09/08						
ENTREGA DE TRABAJO DE GRADUACION AL LECTOR 03/09/08						

Tabla1: Plan de acción de la ejecución del proyecto.

AÑO 2008																																											
N°	ACTIVIDADES	OBJETIVOS	RESPONSABLE	MES 1					MES 2					MES 3					MES 4					MES 5					MES 6					MES 7					MES 8				
				1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
11	Efectuar una investigación sobre los protocolos que permiten el establecimiento de la comunicación Voce sobre Ip: -H323 -SIP -IAX2 por medio de libros, revistas especializadas, internet y personas involucradas con el tema.	Detallar las características principales de los protocolos que permiten la comunicación voz Ip	Sandra Hernández																																								
12	REVISIÓN DEL CAPITULO II POR PARTE DEL ASCESOR																																										
13	CORRECCIÓN DEL CAPITULO II																																										
14	ENTREGA DEL CAPITULO II AL LECTOR																																										
15	REUNION CON EL ASCESOR PARA DEFINIR INDEX CAP II																																										
16	Realizar una investigación monográfica sobre las generalidades del protocolo SIP que incluya: -Historia -Estructura -Funcionamiento -Aplicaciones Por medio de libros, revistas especializadas, internet y personas involucradas con el tema.	Explicar en detalle los aspectos principales del protocolo SIP, que incluya la historia, estructura, funcionamiento, problemas y aplicaciones.	Rodrigo Castro																																								
17	Realizar una investigación y recopilar de manera específica el funcionamiento de los elementos y agentes que permiten el funcionamiento de una red SIP.	Detallar el funcionamiento de cada uno de los elementos que conforman una red SIP	Sandra Hernández																																								
18	REVISIÓN DEL CAPITULO III POR PARTE DEL ASCESOR																																										
19	CORRECCIÓN DEL CAPITULO III																																										
20	ENTREGA DEL CAPITULO III AL LECTOR																																										
21	REUNION CON EL ASCESOR PARA DEFINIR INDEX CAP IV																																										
22	Realizar una investigación monográfica sobre las generalidades Asterisk que incluya: Historia Estructura Capacidades Limitaciones de la arquitectura Aplicaciones de Asterisk	Explicar en detalle los aspectos principales del Asterisk, que incluya la historia, estructura, funcionamiento, problemas y aplicaciones.	Rodrigo Castro, Sandra Hernández																																								
23	REVISIÓN DEL CAPITULO IV POR PARTE DEL ASCESOR																																										
24	CORRECCIÓN DEL CAPITULO IV																																										
25	ENTREGA DEL CAPITULO IV AL LECTOR																																										
26	REUNION CON EL ASCESOR PARA DEFINIR INDEX CAP V																																										

AÑO 2008																																																
N°	ACTIVIDADES	OBJETIVOS	RESPONSABLE	MES 1					MES 2					MES 3					MES 4					MES 5					MES 6					MES 7					MES 8									
				1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5					
27	Seguindo información recopilada con anterioridad sobre asterisk y SIP se redactara la documentación que detalle los elementos necesarios y requerimientos mínimos de cada dispositivo, para iniciar la configuración del servidor.	Generar la información previa necesaria, para lograr llevar acabo la configuración de la red SIP en un entorno Linux. Proporcionar una guía de desarrollo que facilite la implementación de la comunicación voip a partir del servidor Asterisk y protocolo SIP	Sandra Hernández Rodrigo Castro																																													
28	Teniendo en cuenta la actividad anterior se realizara la configuración del software Asterisk bajo el protocolo SIP, en un servidor Linux.	Obtener un servidor en Asterisk configurado con el protocolo SIP, que trabaje de manera optima	Sandra Hernández																																													
VACACIONES AGOSTINAS																																																
29	Realizar pruebas de comunicación del servidor Asterisk, entre los diferentes elementos de la red como son: Computadoras, teléfonos IP, teléfonos análogos.	Verificar el funcionamiento del servidor Asterisk, realizando pruebas de comunicación entre PCs, teléfonos IP y teléfonos análogos, que nos permitan identificar posibles problemas que se pueda presentar.	Rodrigo Castro																																													
30	Con las pruebas realizadas y corrección de errores que puedan surgir, se dará una lista de buenas practicas de construcción y configuración de una red VoIP, que trabaje utilizando el protocolo SIP.	Proporcionar una guía de buenas practicas para el funcionamiento ideal de una red de voz sobre IP a través del protocolo SIP	Sandra Hernández, Rodrigo Castro																																													
31	REDACCIÓN DEL DOCUMENTO FINAL		Sandra Hernández, Rodrigo Castro																																													
32	REVISIÓN DEL CAPITULO Y POR PARTE DEL ASESOR																																															
33	CORRECCION DEL DOCUMENTO		Sandra Hernández, Rodrigo Castro																																													
34	ENTREGA DE TRABAJO DE GRADUACION AL LECTOR		Sandra Hernández, Rodrigo Castro																																													
35	CORRECCION DEL TRABAJO DE GRADUACION		Sandra Hernández, Rodrigo Castro																																													

Tabla2: Cronograma de la ejecución del proyecto.

1.13 Plan Capitular

CAPITULO II. INTRODUCCIÓN A LA TECNOLOGÍA VoIP

2.1 Generalidades de la Telefonía

2.1.1 Evolución histórica de la telefonía.

2.1.2 Funcionamiento de la telefonía Actual.

La conmutación de circuitos.

La conmutación de paquetes

2.1.3 Características de la telefonía.

2.2 Descripción de la tecnología VoIP.

2.2.1 Historia y generalidades de las tecnologías de VoIP

2.2.2 Funcionamiento de los sistemas de VoIP

2.2.2.1 Proceso de Codificación de Voz

Muestreo

Cuantificación.

Cuantificación Uniforme

Cuantificación No Uniforme

Codificación de la Voz

2.2.2.2 Códec utilizados en VoIP

G.711

G723

G729A

2.2.3 Protocolos de soporte de VoIP

2.2.3.1 Protocolos De Señalización VoIP.

H323

SIP

IAX2

2.2.3.2 Protocolos De Transporte

2.2.3.3 Real Time Protocol RTP

2.2.3.4 Real Time Control Protocol RTCP

2.2.3.5 Real Time Streaming Protocol RTSP

2.2.4 Tecnologías Comerciales Existentes De VoIP

2.2.5 Características Y Limitaciones De VoIP

Características

Limitantes

CAPITULO III. PROTOCOLO SIP

3.1 Historia de SIP

3.2 Funcionalidades de SIP

3.2.1 Establecimiento, modificación y terminación de sesión.

3.2.2 Movilidad del usuario.

3.2.2.1 URLs SIP

3.2.2.2 Registro

3.3 Elementos SIP

3.3.1 Agentes de Usuario (UA)

3.3.2 Servidor SIP

3.3.2.1 Proxy SIP

3.3.2.2 Servidor de Localización

3.3.2.3 Servidor de re direccionamiento SIP

3.3.2.4 Servidor de Registro

3.4 Operación De SIP

3.4.1 Mensajes SIP.

3.4.1.1 Mensaje De Solicitud.

3.4.1.2 Mensaje De Respuesta

3.4.1.3 Cabecera De Mensaje SIP

3.4.1.4 Cuerpo De Mensaje SIP

3.4.2 Autenticación

3.4.3 Dificultades Con Los Procesos De
Traducción De Direcciones NAT.

3.4.4 Operación De SIP En La Capa De Transporte

3.4.4.1 Transacción Invite

3.4.4.2 Transacción Cancel

3.4.4.3 Otras Transacciones

3.5 Aplicaciones De SIP

3.6 Ejemplo De Comunicación SIP

CAPITULO IV SOLUCIÓN DE VoIP UTILIZANDO ASTERISK

4.1 Definición y Generalidades

4.2 Historia de Asterisk

4.3 Características De Asterisk

4.3.1 Características Generales

4.3.1 Licenciamiento De Asterisk

4.3.2 Funcionalidades De Llamada

4.3.4 Ventajas

- 4.4 Arquitectura De Asterisk
 - 4.4.1 Directorios Utilizados Por Asterisk
 - 4.4.2 Esquema De La Arquitectura De Asterisk
 - 4.4.2.1 Archivos de aserisk
 - 4.4.2.2 Canales
 - 4.4.2.3 Codecs
 - 4.4.2.4 Formatos
 - 4.4.2.5 Aplicaciones
 - 4.4.2.6 Núcleo PBX
 - 4.4.3 Dial Plan
 - 4.4.4 Función Dial ()
 - 4.4.5 Limitaciones De La Arquitectura De Asterisk
- 4.5 Asterisk Y La Telefonía Tradicional
- 4.6 VoIP Y Asterisk
- 4.7 Utilización De Asterisk A Través De SIP
- 4.8 Aplicaciones De Asterisk
 - 4.8.1 Usos
 - 4.8.2 Aplicaciones

CAPITULO V: GUÍA DE IMPLEMENTACIÓN DE UN SERVIDOR ASTERISK COMO UNA SOLUCIÓN DE VOZ SOBRE IP.

- 5.1 Elementos a utilizar
- 5.2 Requerimientos mínimos de hardware y software
- 5.3 Desarrollo de la guía.
 - 5.3.1 Paso 1 Instalación de Asterisk
 - 5.3.2 Paso 2. Configuración de Asterisk vía consola

5.3.3 Paso 3. Configuración de los clientes SIP

- Configuración de Teléfono IP Polycom IP 301 SIP.
- Configuración del dispositivo adaptador analógico VoIP Solution MTA V 102 .
- Instalación y Configuración De Softphone X-Lite.

5.4 Establecimiento de la llamada Asterisk.

1.14 Presupuesto

A continuación se detallan los costos en los que se incurrirán durante el desarrollo del proyecto, que esta planificado para realizarse en un periodo de 6 meses.

1.14.1 Inversión

Detalle de la inversión:

1. Personal: el proyecto será elaborado por dos personas, a quienes se le cancela los siguientes honorarios.

Nº de personas	Profesión	Sueldo Mensual (\$)	Cantidad de meses	Total (US \$)
2	Ing. En Computación	\$900.00	6	\$5400.00

Tabla 3 Honorarios de la persona a desarrollar el proyecto

2. Papelería y útiles: para poder realizar eficientemente el trabajo, se necesita de papelería y herramientas indispensables como lo son el impresor y tinta. En la categoría de otros, se refiere a gastos de empastado, y gastos que pueden aparecer de improviso durante el desarrollo del proyecto.

Cantidad	Descripción	Precio unitario(\$)	Total(US \$)
5	Resmas de Papel Bond	\$3.90	\$19.50
1	Impresor	\$50.00	\$50.00
5	Cartuchos de Tinta	\$12.00	\$60.00
	Otros	\$300.00	\$300.00
Total			\$429.50

Tabla 4 Papelería y útiles a utilizar en el proyecto

En la tabla 5 se detallan los costos de energía eléctrica, Internet, y transporte.

Descripción	Costo por mes(\$)	Cantidad de meses	Total (US \$)
1. Energía eléctrica	\$40.00	6	\$240.00
2. Internet	\$40.68	6	\$244.00
3. Transporte	\$36.00	6	\$216.00

Tabla 5 Otros gastos considerados en el desarrollo del proyecto.

3. Costos de Hardware

Descripción	Costos (\$)	Cantidad de equipo	Total (US \$)
1. Computadoras	\$600.00	3	\$1800.00
2. Adaptadores telefónicos analógicos (ATA)	\$70.00	1	\$70.00
3. Switch	\$120.00	1	\$120.00
4. Teléfonos analógicos	\$25.00	1	\$25.00
Total			\$2015.00

Tabla 6 Gasto de Equipo para el desarrollo del proyecto

3. Costos de Software.

El software a utilizar es de distribución Abierta por lo cual no incurrirá en costo.

Descripción	Costos (\$)	Cantidad de equipo	Total (US \$)
1. Sistema Operativo Linux, AsteriskNow	\$00.00	1	\$00.00
2. Softphone Xten lite	\$00.00	1	\$00.00
3. Asterisk	\$00.00	1	\$00.00
Total			\$00.00

Tabla 7 Gasto de Software.

CAPITULO II

INTRODUCCIÓN A LA TECNOLOGÍA VoIP

Introducción

La comunicación es una de las actividades más importante para el ser humano, esta permite que las personas puedan expresar sus ideas y necesidades en cualquier lugar o circunstancia; y es así como el hombre se ha esforzado por ir mejorando y facilitando las técnicas de comunicación.

En un inicio el primer medio de comunicación fueron las señales o gestos transmitidos que pronto demostraron su deficiencia ante la oscuridad, por lo cual se vieron en la necesidad de evolucionar a la transmisión de sonidos, hasta lograr transmitir sin dificultad sus ideas; de esta manera, es como el hombre logro llegar a la tecnología de la telefonía, siendo este uno de los mas grandes logros alcanzados en esta área.

El uso de redes IP para la transmisión de voz, representa uno de los avances más significativos desarrollados en los últimos años, lo que apunta a convertirse en la tecnología que predominara el mercado telefónico, debido a que permite en un mismo medio la transmisión de la voz al mismo tiempo que se transmite los datos.

Este capítulo presenta los aspectos básicos de la comunicación telefónica tradicional, al mismo tiempo describe los cambios en la evolución que ha experimentado hasta llegar a detallar el funcionamiento y las características de la reciente tecnología VoIP. Dentro de las características de VoIP se hace especial énfasis en dos aspectos de esta tecnología: el Proceso de Codificación de Voz y los protocolos utilizados en esta; con el fin de introducir e identificar al lector con los términos de la tecnología VoIP.

2.1 Generalidades de la Telefonía

El teléfono es un elemento indispensable en la vida moderna del ser humano. Esta herramienta enlaza al mundo desde hace más de un siglo, convirtiéndose en uno de los principales medios de comunicación para la humanidad, que permite acercar en tiempo y espacio a las personas sin importar las distancias.

Las redes telefónicas fijas, que dan servicio a más de 1200 millones de usuarios, están extendidas por todos los países y funcionan todas ellas de una manera similar. Empleando la técnica de conmutación de circuitos y un sistema de señalización común SS7. Al estar interconectadas entre ellas permiten la realización de llamadas entre cualquiera de los puntos de la misma. Un sistema de numeración que relaciona un número con un área geográfica idéntica a cada país.

Es un servicio que ha evolucionado desde sus inicios, desde los primeros experimentos de tráfico de voz hasta un proceso de comunicación Digital que optimiza el uso de los recursos de la red y proporciona calidad al servicio prestado los usuarios.

2.1.1 Evolución Histórica de la Telefonía

En 1857, Antonio Meucci (1808-1889) había inventado una máquina cuyo componente esencial era un elemento vibrador unido a un imán; era el primer aparato telefónico, pero sería Graham Bell (1847-1922) quien, finalmente, tras patentar un aparato semejante en 1876, pasaría a la historia como el verdadero padre del teléfono.

Un año después, 1877, Edwin T. Holmes sugirió la idea de establecer una central telefónica, donde por medio de un teléfono pudiese conectarse con otro cualquiera. La primera conexión telefónica pública, se realizó en Estados Unidos en 1878, gracias a la instalación de una centralita de funcionamiento manual, que hacía posible la distribución de las llamadas entre los usuarios de la red. Desde la centralita

manual, se establecía la conexión a través de una red de clavijas que se introducían en su correspondiente toma. Es así como paso a paso la red telefónica ha experimentado grandes cambios, marcando así, una evolución en todos sus métodos y ha explotado nuevas tecnologías de comunicación.

Algunas evoluciones son mostradas a continuación.

- Telefonía fija o convencional, que hace referencia a las líneas y equipos que se encargan de la comunicación entre terminales telefónicos no portables, y generalmente enlazados entre ellos o con la central por medio de conductores metálicos.
- centralita telefónica de conmutación manual para la interconexión mediante la intervención de un operador/a de distintos teléfonos, creando de esta forma un primer modelo de red.
- Introducción de centrales telefónicas de conmutación automática, constituidas mediante dispositivos electromecánicos, de las que han existido, y en algunos casos aún existen, diversos sistemas (rotatorios, barras cruzadas y otros más complejos).
- Centrales de conmutación automática electromecánicas, pero controladas por computadora.
- Centrales digitales de conmutación automática totalmente electrónica y controlada por ordenador, que permiten multitud de servicios complementarios al propio establecimiento de la comunicación (los denominados *servicios de valor añadido*).
- Introducción de la Red Digital de Servicios Integrados (RDSI) y las técnicas xDSL o de banda ancha (ADSL, HDSL, etc.), que permiten la transmisión de datos a más alta velocidad.

- Tecnología VoIP que consiste en el transporte de Voz sobre redes IP y es la tecnología mas reciente.

Como se observa la red telefónica en un inicio fue análoga o bien conocida también como Red Telefónica Básica (RTB), que permitió el intercambio bidireccional de tráfico de voz en tiempo real, entre diferentes usuarios a través de una red de conmutación de circuitos PSTN: (Public Switched Telephone Network.)

Mas adelante, siguen surgiendo nuevos avances en la mejora de la tecnología de la red telefónica, hasta llegar al desarrollo de las tecnologías de transmisión de voz por paquetes a partir del aparecimiento inicialmente del internet, en donde este se utilizaba principalmente para fines de comunicaciones caseras con personas de cualquier parte del mundo. Pero luego llevó a pensar a los usuarios, tanto finales involucrados con tecnología como a grandes empresas a nivel mundial, en implementar la transmisión de voz sobre sus redes, ya que esto es de gran importancia, no únicamente por el ahorro que significa en las comunicaciones de larga distancia, si no también por la cantidad de posibilidades que se pueden abrir hacia el nuevo mundo de las redes multiservicios.

2.1.2 Funcionamiento de la Telefonía Actual.

Usualmente las personas utilizan para sus comunicaciones vocales, bien la red telefonía pública conmutada (RTC) o a red digital de servicios integrados (RSDI) si bien las comunicaciones a través de redes móviles (GSM o UMTS) se están expandiendo en gran medida y representan un gran negocio, incluso superior a la telefonía fija. Los usuarios accede al servicio de telefonía fijo por medio de bucles de abonados, que comunican las centrales telefónicas con sus domicilios, bien sea residencial o empresarial.

Las empresas por lo general tienen su propia red telefónica convencional, diseñada sobre PBX, que soportan todos los servicios tradicionales.

La telefonía tradicional es llamada también análoga, y para poder realizar una llamada en una red telefónica análoga se necesitan de dos elementos básicos: el hardware y la señalización.

El hardware de una red telefónica, es toda aquella infraestructura entre el proveedor de servicios telefónicos local y nuestra casa u oficina.

Esta infraestructura consiste en:

- ✓ El Aparato Telefónico o Teléfono.
- ✓ El Bucle Local (Local Loop).
- ✓ La Línea Troncal.
- ✓ El Conmutador o Switch de Voz (Central Pública o Privada).

La Señalización de la red es una serie de reglas a seguir entre las centrales, las troncales y los aparatos telefónicos, para que la llamada pueda iniciarse, establecer y completarse.

La red telefónica cumple tareas sencillas para el usuario; a partir de un par de terminales a los cuales el abonado se conecta. Estos cables llegan a una central, que interactúa con el abonado hasta establecer una conversación con otro usuario mediante una matriz de conmutación.

Para que uno o mas usuarios puedan comunicarse entre si; se marcan los números que corresponden al código del usuario, para ello; el teléfono abre el circuito o lazo en forma intermitente, haciendo que la Central lo interprete, estableciendo conexión con el abonado de destino, a esto se le conoce como marcación decádica². Dependiendo del número marcado, la Central se conectará a otras Centrales, o directamente al abonado, si éstos comparten la misma Central. Así, la Central

² *Marcación decádica: consiste en el envío por el teléfono de la información numérica, en forma de pulsos, a la central telefónica automática para que esta le conecte con el teléfono deseado.*

conectada al abonado destino comprueba que esté desocupado y envía al abonado origen un tono de ocupado.

La conmutación es la conexión que realizan los diferentes modos que existe en los diferentes y distancias para lograr un camino apropiado para conectar a los usuarios de una red; la conmutación permite la descongestión entre los usuarios de la red, disminuyendo el tráfico y aumentando el ancho de banda.

Existen dos tipos de conmutación en las redes telefónicas; para la telefonía convencional se utiliza la conmutación por circuitos y la reciente tecnología VoIP utiliza conmutación por paquetes.

La conmutación de circuitos es un método de transferencia de datos que consiste en establecer un circuito dedicado en una red.

Para establecer el contacto y el paso de la información de estación a estación a través de los nodos intermedios, se requieren estos pasos:

1. Establecimiento del circuito: el emisor solicita a un cierto nodo el establecimiento de conexión hacia una estación receptora. Este nodo es el encargado de dedicar uno de sus canales lógicos a la estación emisora (suele existir de antemano). Este nodo es el encargado de encontrar los nodos intermedios para llegar a la estación receptora, y para ello tiene en cuenta ciertos criterios de encaminamiento, coste, etc.

2. Transferencia de datos: una vez establecido el circuito exclusivo para esta transmisión (cada nodo reserva un canal para esta transmisión), la estación se transmite desde el emisor hasta el receptor conmutando sin demoras de nodo en nodo (ya que estos nodos tienen reservado un canal lógico para ella).

3. Desconexión del circuito: una vez terminada la transferencia, el emisor o el receptor indican a su nodo más inmediato que ha finalizado la conexión, y este nodo informa al siguiente de este hecho y luego libera el canal dedicado. Así de nodo en nodo hasta que todos han liberado este canal dedicado.

La conmutación de paquetes: los datos se transmiten en paquetes cortos. Para transmitir grupos de datos más grandes, el emisor trocea estos grupos en paquetes más pequeños y les adiciona una serie de bits de control. En cada nodo, el paquete se recibe, se almacena durante un cierto tiempo y se transmite hacia el emisor o hacia un nodo intermedio en lugar de crear un circuito entre los participantes de la comunicación, utiliza una red IP de datos, donde los paquetes viajan por diferentes rutas para llegar a su destino y únicamente se abre una breve conexión lo suficientemente amplia para enviar un paquete de voz, así, cuando se produce un silencio en una conversación, los paquetes de datos de otras conversaciones o aplicaciones pueden ser transmitidos por la red, lo que implica un uso más eficiente de la misma.

La técnica de la conmutación por paquete consiste cuando un emisor necesita enviar un grupo de datos mayor que el tamaño fijado para un paquete, éste los trocea en paquetes y los envía uno a uno al receptor.

Hay dos técnicas básicas para el envío de estos paquetes:

1. *Técnica de datagramas:* cada paquete se trata de forma independiente, es decir, el emisor enumera cada paquete, le añade información de control (por ejemplo número de paquete, nombre, dirección de destino, etc.) y lo envía hacia su destino. Puede ocurrir que por haber tomado caminos diferentes, un paquete con número por ejemplo 6 llegue a su destino antes que el número 5. También puede ocurrir que se pierda el paquete número 4. Todo esto no lo sabe ni puede controlar el emisor, por lo que tiene que ser el receptor el encargado de ordenar los paquetes y saber los que se han perdido (para su posible reclamación al emisor), y para esto, debe tener el software necesario.

2. *Técnica de circuitos virtuales* : antes de enviar los paquetes de datos , el emisor envía un paquete de control que es de Petición de Llamada , este paquete se encarga de establecer un camino lógico de nodo en nodo por donde irán uno a uno todos los paquetes de datos . De esta forma se establece un camino virtual para todo el grupo de paquetes. Este camino virtual será numerado o nombrado inicialmente en el emisor y será el paquete inicial de petición de llamada, el encargado de ir informando a cada uno de los nodos por los que pase los paquetes de datos con ese nombre o número. De esta forma, el encaminamiento sólo se hace una vez (para la Petición de Llamada). El sistema es similar a la conmutación de circuitos, pero se permite a cada nodo mantener multitud de circuitos virtuales a la vez.

2.1.3 Características de la Telefonía.

✓ Método de conmutación.

La telefonía tradicional para gestionar la llamada hace uso de la conmutación de circuitos, lo que genera altos costos de interconexión si se desea realizar llamadas a grandes distancias geográficas, o si dicha llamada requiere el uso de redes telefónicas que no son propiedad de una misma compañía telefónica.

✓ Calidad de voz.

Las redes telefónicas que se usan actualmente, fueron diseñadas especialmente para el manejo de tráfico de voz, es por esta razón que dificultades como: retardos, interferencias o eco son mínimos.

✓ Alta disponibilidad.

La comunicación telefónica esta regulada para presentar un porcentaje de disponibilidad de un 99.999% lo que se traduce en una falta de servicio de solo 5.25 minutos al año. Razón por la que la telefonía tradicional sigue funcionando, incluso cuando existen fallas de energía o aun en situaciones de emergencia.

✓ Precio.

El cobro de la realización de una llamada se calcula en base al tiempo de conexión y a la distancia entre los usuarios.

2.2 Descripción de la Tecnología IP.

La telefonía IP conjuga dos mundos históricamente separados entre dos tipos de redes; la transmisión de voz y la de datos. Se trata de transportar la voz, previamente convertida a datos, entre dos puntos distantes. Esto posibilita utilizar las redes de datos para efectuar las llamadas telefónicas, y yendo un poco más allá, desarrollar una única red convergente que se encargue de cursar todo tipo de comunicación, ya sea voz, datos, video o cualquier tipo de información

El término Voz sobre IP, VoIP por sus siglas en inglés (Voice Over Internet Protocol), se refiere a la utilización de redes IP y sus protocolos para comunicaciones de voz, permitiendo que la voz al igual que los datos utilice el mismo medio de transmisión.

Se distinguen dos diferencias para la aplicación de VoIP en servicios de telefonía. El primero es cuando la VoIP es transportada a través de redes privadas o empresariales, y el segundo cuando el internet es utilizado para transportar la comunicación de dos extremos. En este caso se conoce como Telefonía por Internet. La diferencia no sólo solo es el medio de transporte sino también en las posibilidades de establecer mecanismos de control (QoS) que garanticen la máxima calidad de servicio en todo momento. Los mecanismos y técnicas aplicadas en ambos casos difieren sustancialmente pero a través de éstas se consiguen niveles de calidad de servicio similares a la telefonía convencional PSTN.

La voz IP, no es en sí mismo un servicio, sino una tecnología que permite encapsular la voz en paquetes para poder ser transportados sobre redes de datos sin necesidad de disponer de los circuitos conmutados convencionales PSTN, (las redes desarrolladas a lo largo de los años para transmitir las conversaciones vocales, se basaban en el concepto de conmutación de circuitos). VoIP no utiliza circuitos para la conversación, sino que envía múltiples de ellas (conversaciones) a través del mismo canal codificadas en paquetes y flujos independientes. Cuando se produce un

silencio en una conversación, los paquetes de datos de otras conversaciones pueden ser transmitidos por la red, lo que implica un uso más eficiente de la misma.

2.2.1 Historia y Generalidades de las Tecnologías de VoIP

Voz sobre IP nace a finales de los Ochenta y principio de los Noventa, cuando se inicio con la idea de integrar todos estos servicios en una sola red para ahorrar costos y aumentar la eficiencia de los mismos. Con el desarrollo de ISDN y redes similares, se hizo posible el envío de voz y datos en una misma red.

Sin embargo, a mediados de los años noventa la red de Internet, debido al desarrollo de sus protocolos y rápida aceptación, demostró ser la red que podía lograr la integración de todos los servicios ya que podía enviar todos los tipos de paquetes multimedia como voz, datos y vídeo; en todos los tipos de red a un menor costo.

2.2.2 Funcionamiento de los Sistemas VoIP

Hace unos años a tras se descubrió que también se podía enviar señales a su destino de manera digital, es decir, antes de enviar la señal se debería digitalizar con un dispositivo ADC (analog to digital converter), transmitirla y en el extremo de destino transformarla de nuevo a formato análogo con un dispositivo DAC (digital to analog converter).

VoIP funciona de esa manera, digitalizando la voz en paquetes de datos, enviándola a través de la red y reconvirtiéndola a voz en el destino. Básicamente el proceso comienza con la señal análoga del teléfono que es digitalizada en señales PCM (pulse code modulación) por medio del codificador/decodificador de voz (codec). Las muestras PCM son pasadas al algoritmo de compresión, el cual comprime la voz y la fracciona en paquetes que pueden ser transmitidos para este caso a través de una red privada WAN. En el otro extremo de la nube se realizan exactamente las mismas funciones en un orden inverso.

2.2.2.1 Proceso De Codificación De Voz

La realización de la comunicación telefónica por medios digitales, hace necesario que la voz, cuya naturaleza es análoga, sea convertida a una señal digital que permita la misma calidad de sonido, y elimine el ruido de una conversación. Este proceso implica el muestreo de la señal (discretización temporal) y la posterior cuantificación de las muestras obtenidas (discretización en amplitud). Posteriormente, las muestras de voz cuantificadas habrán de ser codificadas adecuadamente para su transmisión por el canal de comunicaciones.

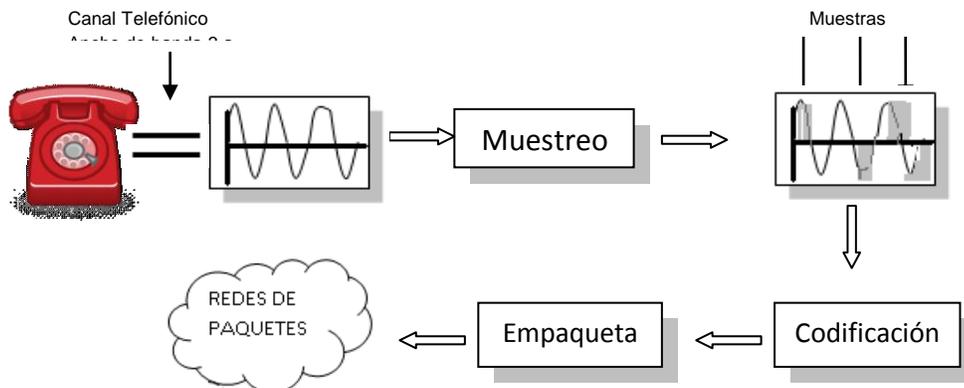


Figura 1: Proceso de codificación de la Voz

Muestreo.

Consiste en seleccionar los valores de la señal análoga en ciertos instantes concretos teniendo especial cuidado en la selección de los valores para evitar tener pérdidas de información, que se supone el muestreo no representa una pérdida importante.

Se puede hacer uso del teorema de Nyquist, que indica que una señal análoga para poder ser reconstruida totalmente de la información digital, esta debe ser muestreada por lo menos al doble de la frecuencia de transmisión más alta.

$$f_s \geq 2 \cdot BW \text{ señal}$$

Así, en el caso de un canal telefónico de 4KHz con las frecuencias de guarda, el muestreo debe de ser a 8000Hz, es decir, una muestra cada 0.125 milisegundos.

En efecto, el muestreo periódico de la señal puede verse, en el dominio de la frecuencia, como una repetición de dicha señal cada F_s . Si el valor de la frecuencia de muestreo es menor que el doble de ancho de banda de la señal, se produce un solapamiento entre versiones desplazadas del espectro que hace imposible la recuperación de la señal original fácilmente.

Cuantificación.

El resultado del muestreo es un conjunto de valores de la señal tomada en ciertos instantes de tiempo, pero la señal sigue siendo continua en amplitud y es necesario discretizarla³ también en este dominio. La cuantificación es el proceso mediante el cual se asignan valores discretos, a las amplitudes de las muestras obtenidas en el proceso de muestreo.

El objetivo de la cuantificación es conseguir que la señal quede representada en un número finito de bits, N , con el que podrá representarse hasta $2^N - 1$ valores diferentes. El resultado será la representación digital de la señal. Para cuantificar la señal, se divide su rango dinámico (conjunto de valores que puede tomar la señal) en ' M ' niveles de tamaño ' q ' (paso del cuantificador o intervalo de cuantificación), de manera que el valor cuantificado vendrá dado según una cierta distribución.

Existen varias formas de cuantificar entre ellos:

- Cuantificación Uniforme
- Cuantificación no Uniforme
- Cuantificación Diferencial

Cuantificación Uniforme

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

³ Discretización: se refiere a traducir una señal analógica a una señal digital.

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

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

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

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

Cuantificación No Uniforme.

Normalmente se utilizan 256 intervalos de cuantificación desiguales, donde los segmentos cercanos a la referencia tienen más intervalos de cuantificación y por lo tanto, están más unidos, de esta forma se evita variaciones grandes en las señales de magnitud pequeña, a medida que los intervalos se alejan de la referencia, se hacen menos en cantidad y más grandes. A este método se le llama *Cuantificación no uniforme*.

Cuantificación Diferencial

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

En la cuantificación diferencial, en lugar de tratar cada muestra separadamente, se cuantifica y codifica la diferencia entre una muestra y la que le precede. Como el número de intervalos de cuantificación necesarios para cuantificar la diferencia entre dos muestras consecutivas es lógicamente inferior al necesario para cuantificar una muestra aislada, la cuantificación diferencial permite una reducción sensible de la frecuencia de transmisión en línea, ya que esta es proporcional al número de intervalos de cuantificación.

Codificación de La Voz

Una vez que la señal ya presenta un formato digital el paso siguiente es codificarla, es decir, adaptarla para que sus características sean las idóneas a la hora de transmitirla por un canal de comunicaciones concreto.

La codificación consiste en asignar un código binario mediante una sucesión de 1's y 0's, a cada uno de los valores discretos de la señal (con k veces modifico M valores, siendo $M = 2$ a la k)

Tradicionalmente, en entornos telefónicos se ha venido utilizando la modulación por codificación de pulso MIC (Modulación por Impulsos Codificados), en la que cada muestra de voz se representa por 8 bits, resultando un flujo de 64 kbps (8.00×8) que coincide con la velocidad asignada aun canal básico de la RDSI.

2.2.2.2 Códec Utilizados En VoIP

La voz es una comunicación analógica, mientras que las redes de datos son digitales, para lograr que la voz viaje sobre Internet esta tiene que ser digitalizada y comprimida, a partir de un proceso que convierte ondas analógicas a información digital con un codificador-decodificador llamado CODEC.

El códec además de ejecutar la conversión de analógico a digital, este comprime la secuencia de datos, y proporciona la cancelación del eco.

La compresión de la forma de onda representada puede permitir el ahorro del ancho de banda. Lo que es interesante en los enlaces de poca capacidad ya que permite tener un mayor número de conexiones de VoIP simultáneamente.

Entre los códec utilizados en la tecnología VoIP se mencionan:

G.711

G.711 fue estandarizado formalmente en 1988, el codec del PCM G.711 es el que permite la mejor calidad de voz en la telefonía digital. Inventado por Bell Systems e introducido en los años 70 tempranos, el tronco digital T1 empleó un esquema de codificación sin comprimir de la modulación de código de pulso de 8 pedacitos con un índice de la muestra de 8000 muestras por segundo. Esto permitió el ancho de banda máximo (teórico) de la voz de a de 4000 hertzios. Un tronco T1 lleva 24 canales digitales del PCM multiplexados juntos. El estándar europeo mejorado E1 lleva 30 canales.

Características Principales

- ✓ G.711, permite una señal de audio de calidad tarifada con un ancho de banda de 3.4 KHz que ha de ser codificado para la transmisión de índices de 56 Kbps o 64 Kbps. El G.711 utiliza A-law o Mu-law para una compresión simple de amplitud y es el requisito básico de la mayoría de los estándares de comunicación multimedia de la ITU.
- ✓ G.711 brinda la mejor calidad de voz; ya que no usa ninguna compresión y es el mismo codec utilizado por la red RTC⁴ y líneas RDSI, suena como si utilizáramos un teléfono RDSI normal. También tiene la menor latencia puesto que no hay necesidad de compresión, lo cual cuesta menos capacidad de procesamiento.
- ✓ Toma un ancho de banda de hasta 84 Kbps incluyendo todo el TCP/IP de arriba. Sin embargo, con el aumento de bandwidth de banda ancha, esto no debe ser un problema.

⁴ RTB: Red Telefónica Conmutada

Sub-divisiones

El codec g711 tiene dos versiones conocidas como alaw (usado en Europa) y ulaw (usado en USA y Japón). U-law se corresponde con el estándar T1 usado en Estados Unidos y A-law con el estándar E1 usado en el resto del mundo. La diferencia es el método que se utiliza para muestrear la señal. La señal no se muestrea de forma lineal sino de forma logarítmica. A-law tiene un mayor rango

- ✓ **G.711 U-law:** El algoritmo Ley Mu (μ -law ó mu-law) es un sistema de cuantificación logarítmica de una señal de audio. Es utilizado principalmente para audio de voz humana. Este sistema de codificación es usado en EEUU y el Japón.
- ✓ **G.711 A-law:** La **ley A** (A-Law) es un sistema de cuantificación logarítmica de señales de audio, usado habitualmente con fines de compresión en aplicaciones de voz humana. Está estandarizada por la ITU-T. Este sistema de codificación es usado en Europa.

Proceso De Codificación Decodificación Utilizado

G.711 A-law

El algoritmo Ley A basa su funcionamiento en un proceso de compresión y expansión llamado companding. Se aplica una compresión/expansión de las amplitudes y posteriormente una cuantificación uniforme. Las amplitudes de la señal de audio pequeñas son expandidas y las amplitudes más elevadas son comprimidas.

G.711 U-law:

La implementación del sistema consiste en aplicar a la señal de entrada una función logarítmica y una vez procesada realizar una cuantificación uniforme. Es lo mismo que decir que el paso de cuantificación sigue una función del tipo logarítmico.

G.723

G.723 es un estándar ITU-T de codec de voz de banda ancha. Esta es una extensión de acuerdo a la recomendación G.721 adaptiva del pulso diferencial del código de modulación de 24 y 40 kbit/s para equipos de aplicaciones de multiplicación de circuitos digitales supercedido por el G.726, este estándar es obsoleto.

Características Principales

- ✓ Un standard ITU standard del tipo narrow-band audio codec que codifica el habla en una cadena de datos cada 30ms (240 muestras en total).
- ✓ Cada frame puede ser de 24 o 20 bytes de longitud, lo que hace a la cadena de datos tanto de 6.4kb/sec o 5.3kb/sec.
- ✓ Este códec esta cubierto por una variedad de patentes, lo que significa que debe ser pagada una patente antes de poder ser utilizado comercialmente.
- ✓ G723 consigue una baja tasa de bit, reduciendo considerablemente la necesidad de ancho de banda, por lo que resulta apropiado para internet
- ✓ El códec de audio G.723.1 se usa mayoritariamente en aplicaciones VoIP y necesita una licencia para poder usarse. G.723.1 es un tipo de códec de alta calidad y de compresión elevada.

Sub-Divisiones

G.723.1. Apareció en noviembre de 1988, incluye un esquema de compresión basado en la supresión de silencios y VAD (Voice activity Detection). Estos esquemas de codificación intentan reproducir el sonido subjetivo de la señal, más que la forma de onda, pero son muy sensibles a la pérdida de paquetes o al jitter, por lo que emplean, por ejemplo, las técnicas de interpolación para reducir el mínimo estos efectos.

G.723.1 define cómo puede codificarse una señal de audio con un ancho de banda de 3.4 KHz para transmitirse a 5.3 Kbps y 6.4 Kbps. Este codec requiere un índice de transmisión muy bajo ofreciendo una calidad de audio cercana a la tarificada. G.723.1 ha sido seleccionada por el VoIP Forum como el codec básico para aplicaciones de telefonía IP de bajo índice de bits.

G.729

EL G.729 es un algoritmo de compresión de datos de audio para la transmisión de voz, este es un codec propietario altamente robusto pero requiere de una licencia para su uso comercial. G729 es un códec de 8Kbps (aprox. 30Kbps por conversación usando SIP) el códec fue desarrollado por un consorcio de organizaciones: France Telecom, Mitsubishi Electric Corporation, Nippon Telegraph and Telephone Corporation (NTT) y a Universidad de Sherbrooke. Y es usado mayoritariamente en aplicaciones de Voz sobre IP por sus bajos requerimientos en ancho de banda.

Características Principales

- ✓ G.729 y G.729A tienen una latencia (el tiempo que necesita para convertir de analógico a digital) más baja que G.723.1
- ✓ El tamaño de una trama ACT⁵ es de 80 bits en el códec G729
- ✓ G729 tiene un costo; es decir no es libre y la calidad de voz es baja

Sub-Divisiones

- ✓ G729A. Anexo A (aprobado en noviembre de 1995). Es una simplificación de la versión original G729 y es compatible con G729. Este requiere cerca de 10.5 MIPS para codificación y 2 MIPS para decodificación.

⁵ Trama ATC son las tramas que llevan información de la voz comprimida

- ✓ G729B o anexo B: Es G729 pero con supresión de silencios y no es compatible con las anteriores.
- ✓ G729AB: Es G729A con supresión de silencios y sería compatible solo con G729B.

Proceso de Codificación decodificación utilizado

El códec G729 trabaja comprimiendo la voz de audio en trozos de 10 milisegundos. La música o los tonos tales como los tonos de DTMF (Multifrecuencia de doble tono) o de fax no pueden ser transportados confiablemente con este códec.

Recientemente, G.729 se ha ampliado para prestar apoyo a la ancha expresión y la codificación de audio, es decir, la gama de frecuencias de transmisión acústica se amplía a 50 Hz - 7 kHz.

2.2.3 Protocolos de Soporte de VoIP

2.2.3.1 Protocolos de Señalización VoIP.

Antes de realizar la comunicación en una red de voz sobre redes IP, es necesario que a las terminales involucradas se les reserven recursos del sistema, establezcan sesión, sincronicen la comunicación, y establezcan una comunicación permanente, que durara hasta que uno de los participantes decida terminar la conexión. Las expectativas de calidad de servicio al igual que la telefonía tradicional son altas, por lo que se hace necesario seleccionar un protocolo de señalización adecuado cuya arquitectura de red cumpla con dichas expectativas.

H323

Es un conjunto de protocolos que definen los elementos y los medios de interacción entre los mismos, que deben de cumplirse para permitir comunicación multimedia en redes de paquetes sin garantías de servicio como las redes IP. Fue desarrollado en

1996 bajo el organismo de estandarización ITU, para soportar comunicación multimedia sobre redes LAN, posteriormente se aplicó al transporte de voz sobre redes IP, y actualmente gran parte de gateways y gatekeepers en el mercado lo soportan.

Elementos Del Protocolo H.323

Terminales: Son los equipos utilizados por los usuarios y abarcan desde teléfonos tradicionales hasta teléfonos IP, al igual que computadoras equipadas con tarjetas de sonido y micrófono. Estos equipos permiten principalmente la comunicación de audio y en algunos casos pueden realizar comunicación de video.

Gatekeepers: Es el elemento más crítico de una red H.323 debido que se encarga de las tareas de control en una zona de influencia. Dentro de esa zona, se encarga de ofrecer servicios a entidades registradas, entre los que se encuentran: gestión de zona, el control de admisión, la gestión de ancho de banda y la traducción de direcciones (entre IP y número de teléfono).

Gateways: Estos elementos se encargan de interconectar una red H.323 con otra que no lo sea. Sus funciones básicas son la traducción de protocolos de establecimiento, liberación de llamadas y la conversión de los formatos de la información entre diferentes tipos de redes.

Unidad multiconferencia: son dispositivos empleados en comunicaciones simultáneas entre más de dos usuarios. Se encargan de mezclar los flujos de audio y video al mismo tiempo de distribuir dichos flujos entre los participantes.

Arquitectura del Protocolo H.323

El desarrollo de una red de comunicación de voz IP sobre el protocolo de señalización H.323 requiere principalmente de dos elementos básicos; terminales y el Gatekeepers. Las terminales permiten representar un dispositivo de comunicación

que bien pueden ser: teléfonos IP, SoftPhone o aun teléfonos tradicionales con adaptadores análogos digitales. Los gatekeepers por su parte, representan un punto de unión entre los participantes, y es el que se encarga de gestionar el proceso de comunicación.

En la estructura del protocolo H.323 se pueden apreciar los diferentes elementos que lo integran, ya que cada uno cumple con una función específica. En el caso de los flujos de video y audio los protocolos encargados de realizar dicha labor son RTP y RTCP, mientras el flujo de datos se realiza a través de la recomendación T.12x; y para el nivel de transporte pueden utilizar TCP o UDP.

Antes de lograrse la comunicación entre los elementos interesados, es necesario que las terminales se comuniquen con el gatekeeper, en este proceso se utiliza la especificación H.225 conocida como Registration, Admission and Status (SER). Esta comunicación tiene como objetivo realizar el registro, la admisión de de comunicación, y el control del estado de una terminal perteneciente a la red H.323. Para cada llamada , es necesario establecer un canal de señalización entre cada terminal y el gatekeeper a través de un canal RAS, para la señalización entre las terminales involucradas se utiliza un canal bajo la norma Q.931, y un canal lógico de control entre los terminales se realiza utilizando la recomendación H.245.

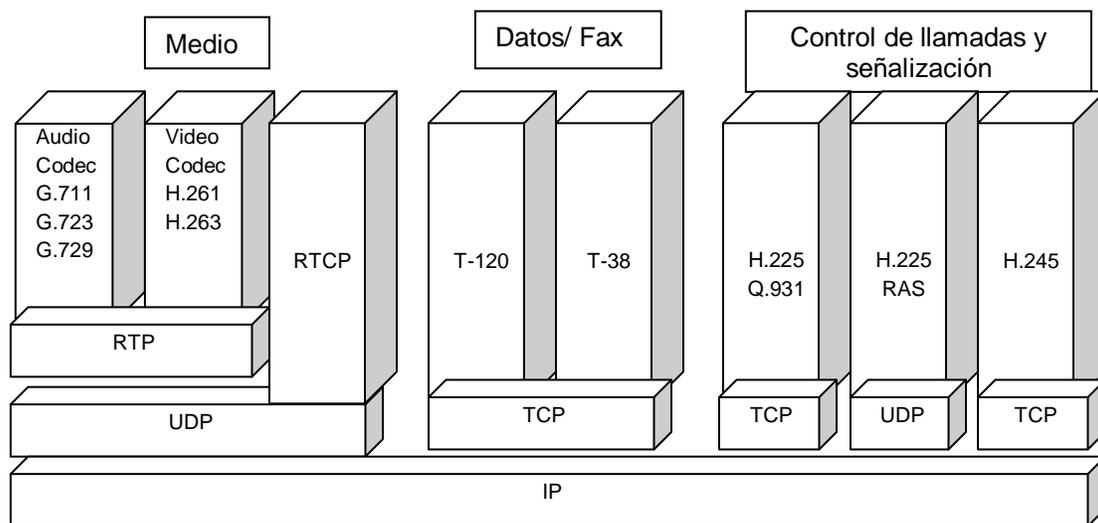


Figura 2: Arquitectura del protocolo H.323

SIP

En marzo de 1999 el IETF publica el protocolo de inicio de sesión SIP bajo la norma RFC 2543. SIP es un protocolo de control de la capa de aplicación que define como establecer, modificar o finalizar una sesión entre dos o más extremos, independientemente del tipo de sesión de que se trate.

El estándar SIP solo define los elementos que participan en un entorno SIP y el sistema de mensajes que intercambian entre ellos. Estos mensajes están basados en http y se emplean esencialmente en procedimientos de registro y para establecer entre que direcciones IP y puertos TCP/UDP intercambiaran datos los usuarios.

La suite de protocolos que trabajan junto a SIP en servicios de VoIP es muy sencilla, utiliza como protocolo de transporte RTP, codificación RTCP, para el transporte utiliza RTCP, y G.711 para compresión.

Elementos de La Red SIP

Agentes de Usuario (UA, User Agent): Son los terminales de los que parten las solicitudes de iniciar una nueva llamada o de terminar una llamada en curso. Se dividen a nivel lógico en dos entidades

Agente de Usuario Cliente: que se encarga de iniciar sesiones SIP.

Agente de usuario de Servidor: es el responsable de aceptar las peticiones de sesión recibidas.

Ambas partes, pueden terminar una sesión en curso, pueden ser teléfonos IP o Softphones.

Servidor Proxy: es una entidad intermedia que actúa como servidor y como cliente y que realiza peticiones de parte de otros clientes. Estas peticiones son servidas internamente o reenviadas a otros servidores.

Servidor de localización: proporciona información acerca de la localización del usuario. Si un usuario A desea comunicarse con un usuario B, en primer lugar A necesita descubrir la localización actual de B en la red, con el fin de que la petición de establecimiento de sesión pueda llegarle. Además, hay que tener en cuenta que el usuario B puede estar en diferentes lugares en instantes distintos, incluso ser alcanzable por varios medios simultáneamente.

Servidor de redirección: acepta una petición SIP, mapea la dirección en cero o más direcciones nuevas y devuelve estas direcciones al cliente. A diferencia de los servidores proxy, los servidores de redirección no inician su propia petición SIP. Tampoco pueden aceptar o terminar llamadas como ocurren con los UAS.

Servidor de registro: Acepta peticiones de registro de los UAC y actualiza la información relativa a cada uno de ellos en una base de datos de localización.

Arquitectura del Protocolo SIP.

La arquitectura de SIP es similar a la de http, en el sentido en que las solicitudes del cliente son enviadas a un servidor que procesa la información para luego enviar una respuesta al cliente. Cuando un usuario desea establecer una comunicación.

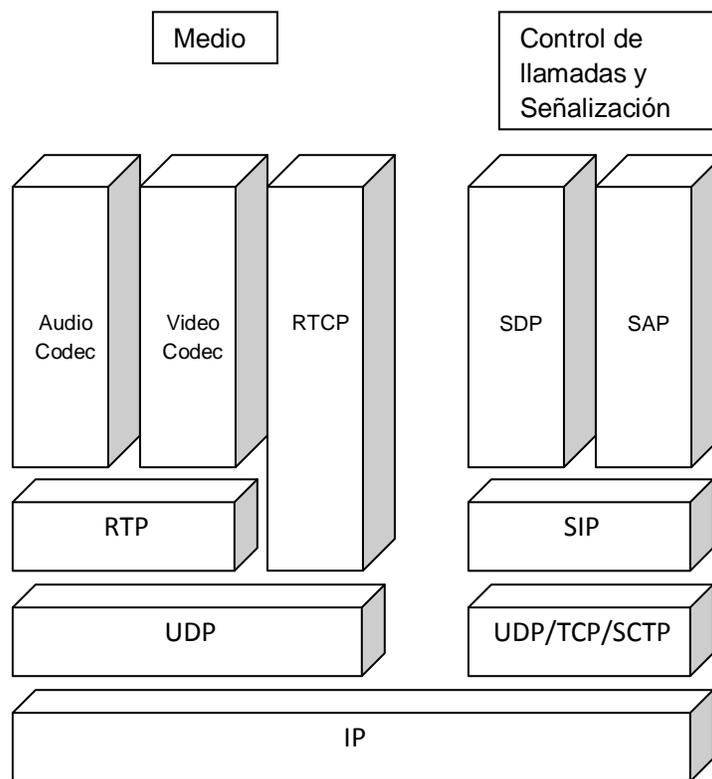


Figura 3: Arquitectura del protocolo SIP

Inter Asterisk Exchange IAX2

Es un protocolo creado y estandarizado por Asterisk. Unas de sus principales características son: *Media* y *señalización* viajan en el mismo flujo de datos.

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

2.2.3.2 Protocolos de Transporte

El objetivo de cualquier comunicación es el intercambio de información entre un elemento origen y un elemento destino, el transmitir esta información de un punto a

otro es la función de los protocolos de transporte, además de fragmentar y reensamblar los paquetes de datos, y proveer los mecanismos necesarios para reducir el impacto de las pérdidas, el jitter, el retardo, etc.

Real Time Protocol RTP

Es el estándar que define las comunicaciones de audio y video en tiempo real sobre redes IP. Proporciona funciones de transporte extremo a extremo y ofrece servicios tales como identificación del tipo de carga y numeración de secuencia. No garantiza la entrega de tráfico en tiempo real pero si suministra los recursos para que se entreguen de manera sincronizada.

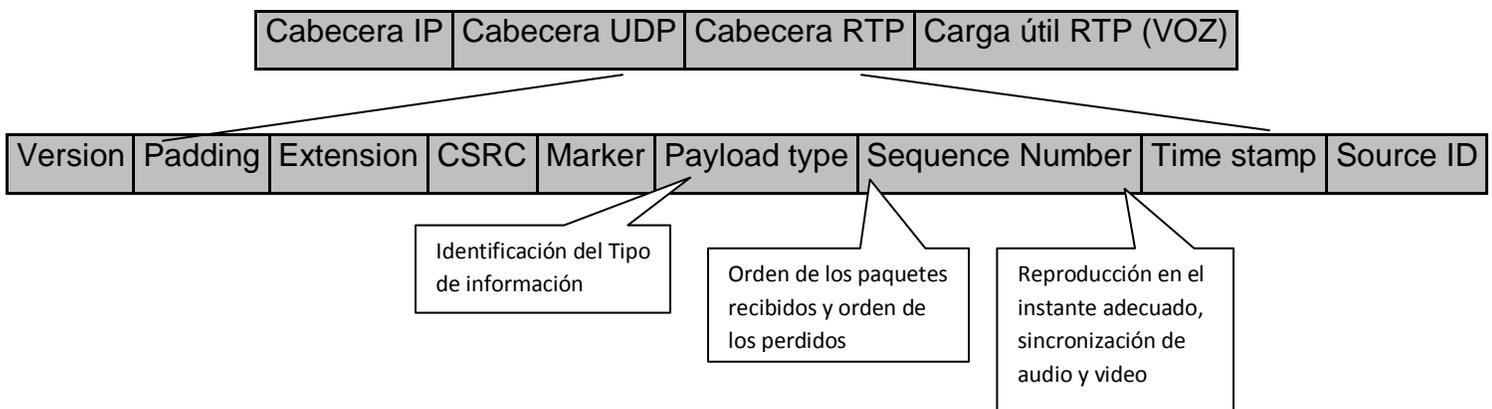


Figura 4: Campos del protocolo RTP

RTP permite identificar el tipo de información transportada, añadir marcas temporales, numero de secuencias y controlar la llegada de los paquetes. Esta información es utilizada por los receptores para reconstruir el flujo de paquetes que género el emisor, eliminando en la medida de lo posible, los efectos de las pérdidas, el retardo y el jitter.

Existe una versión de RTP llamada CRTP que ofrece la posibilidad de comprimir las cabeceras utilizadas por RCT, pero al utilizar esta versión se introduce un retardo adicional debido al proceso de compresión.

Real Time Control Protocol RTCP

RTCP Describe el intercambio de mensajes de control relacionados, fundamentalmente con la calidad de servicio (retardo jitter, tasa de pérdidas, etc.). Aunque es opcional, su utilización es recomendable porque proporciona información de estado de la comunicación con el fin de detectar situaciones en las que la calidad de transmisión no es suficiente, aunque no provee de mecanismos necesarios para mejorar las prestaciones de red.

Real Time Streaming Protocol RTSP

Es un protocolo de nivel de aplicación que define como debe llevarse a cabo el streaming. Se entiende streaming como la capacidad de distribución de contenido multimedia de manera que es posible visualizarlos mientras están siendo transmitidos.

Una vez que la aplicación cliente ha recibido suficientes paquetes comienza la reproducción y simultáneamente, puede estar descomprimiendo otro. Por otra parte un servidor mantiene información de estado de cada cliente que este conectado a el.

2.2.4 Tecnologías Comerciales Existentes de VoIP

La utilización de servicios de VoIP a incrementado considerablemente durante los últimos años, por esta razón han surgido compañías que tienen como objetivo brindar este servicio, otras en cambio se están dedicando al desarrollo de equipo, aplicaciones, y software que trabaje específicamente con tecnología VoIP.

La gran mayoría de software de mensajería instantánea en la actualidad incluyen la implementación de servicios de VoIP, servicios como: Live Messenger, Google Talk, Yahoo Messenger, entre otros. Empresas desarrolladoras de equipo de comunicación como cisco, 3com, Avaya, Ericsson, Mitel, Nortel, Siemens; disponen actualmente de la tecnología necesaria para implementar una red VoIP, proporcionan aparatos telefónicos IP, adaptadores telefónicos analógicos, dispositivos intermedios

de red y centrales telefónicas. Todos estos dispositivos funcionan específicamente para la comunicación de VoIP.

2.2.5 Características Y Limitaciones de la VoIP

Características

- ✓ Seguridad: Internet es de naturaleza insegura para compartir datos. Sin embargo, SIP puede encriptar y autenticar mensajes de señalización, y RTP soporta encriptación del medio transportado. En forma conjunta, pueden entregar una comunicación segura.

- ✓ En la mayoría de las empresas, suelen ser necesarios dos tipos de cableados. Uno para la red de datos a la que se unen los ordenadores, y otro para la red telefónica, a la que se unen los teléfonos. Una de las ventajas de utilizar VoIP sobre una LAN, es el ahorro del cableado necesario para el teléfono. Utilizando VoIP pueden unirse ambas redes internas en una sola; la red de datos: sobre la cual los ordenadores pueden seguir enviando su información como siempre. Pero además es posible aprovechar el ancho de banda no utilizado para enviar voz, de tal forma que las comunicaciones internas se realizarían mediante VoIP, usando el software adecuado en los ordenadores, o mediante teléfonos IP implementados directamente en hardware, que se conectarían a la red, como si de otro dispositivo normal se tratara.

- ✓ VoIP tiene como base la tecnología denominada Packet Switched en la que son encapsulados datos para ser distribuidos a lo largo de un medio compartido. La ventaja real de esta tecnología es la transmisión de voz de forma gratuita, ya que viaja como datos.

- ✓ La tecnología voz sobre IP permite el ahorro de dinero con llamadas que van fuera de la red interna de una empresa.

- ✓ Con VoIP uno puede realizar una llamada desde cualquier lado que exista conectividad a internet. Dado que los teléfonos IP transmiten su información a través de internet estos pueden ser administrados por su proveedor desde cualquier lugar donde exista una conexión. Esto es una ventaja para las personas que suelen viajar mucho, estas personas pueden llevar su teléfono consigo siempre teniendo acceso a su servicio de telefonía IP.

- ✓ Un servicio de VoIP incluye:
 - Identificación de llamadas.
 - Servicio de llamadas en espera
 - Servicio de transferencia de llamadas
 - Repetir llamada
 - Devolver llamada
 - Llamada de 3 líneas (three-way calling).

- ✓ En base al servicio de identificación de llamadas existen también características avanzadas referentes a la manera en que las llamadas de un teléfono en particular son respondidas. Por ejemplo, con una misma llamada en Telefonía IP puedes:
 - Desviar la llamada a un teléfono particular
 - Enviar la llamada directamente al correo de voz
 - Dar a la llamada una señal de ocupado.
 - Mostrar un mensaje de fuera de servicio

- ✓ Aun hoy en día existen problemas en la utilización de VoIP, queda claro que estos problemas son producto de limitaciones tecnológicas y se verán solucionadas en un corto plazo por la constante evolución de la tecnología.

- ✓ VoIP requiere de una conexión de banda ancha. Aun hoy en día, con la constante expansión que están sufriendo las conexiones de banda ancha todavía hay hogares que tienen conexiones por modem, este tipo de conectividad no es

suficiente para mantener una conversación fluida con VoIP. Sin embargo, este problema se vera solucionado a la brevedad por el sostenido crecimiento de las conexiones de banda ancha.

- ✓ VoIP es susceptible a virus, gusanos y hacking, a pesar de que esto es muy raro y los desarrolladores de VoIP están trabajando en la encriptación para solucionar este tipo de problemas.
- ✓ El uso de esta tecnología es amplio. Se la puede utilizar en forma de equipos especializados para actuar como PBX (Private Branch Exchange) o conmutadores, software para instalarse en servidores de datos, o en aplicaciones tipo Messenger para uso personal en PC.
- ✓ Por su estructura el VoIP proporciona las siguientes ventajas. En principio, permite el control del tráfico de la red, por lo que se disminuyen las posibilidades de que se produzcan caídas importantes en el rendimiento. Además, al estar la información comprimida, se puede pasar más volumen en el mismo ancho de banda. El hardware es más sencillo, no se necesita tener conocimientos tan específicos como los que se necesitan para telefonía. Y por último, es más barato.

Limitantes

- ✓ VoIP requiere de una conexión eléctrica. En caso de un corte eléctrico a diferencia de los teléfonos VoIP los teléfonos de la telefonía convencional siguen funcionando (excepto que se trate de teléfonos inalámbricos). Esto es así porque el cable telefónico es todo lo que un teléfono convencional necesita para funcionar.
- ✓ Llamadas al 911: Estas también son un problema con un sistema de telefonía VoIP. Como se sabe, la telefonía IP utiliza direcciones IP para identificar un numero telefónico determinado, el problema es que no existe forma de asociar una

dirección IP a un área geográfica, como cada ubicación geográfica tiene un número de emergencias en particular no es posible hacer una relación entre un número telefónico y su correspondiente sección en el 911.

- ✓ Dado que VoIP utiliza una conexión de red la calidad del servicio se ve afectado por la calidad de esta línea de datos, esto quiere decir que la calidad de una conexión VoIP se puede ver afectada por problemas como la alta latencia (tiempo de respuesta) o la pérdida de paquetes. Las conversaciones telefónicas se pueden ver distorsionadas o incluso cortadas por este tipo de problemas. Es indispensable para establecer conversaciones VoIP satisfactorias contar con una cierta estabilidad y calidad en la línea de datos.
- ✓ En los casos en que se utilice un softphone la calidad de la comunicación VoIP se puede ver afectada por la PC, digamos que estamos realizando una llamada y en un determinado momento se abre un programa que utiliza el 100% de la capacidad de nuestro CPU, en este caso crítico la calidad de la comunicación VoIP se puede ver comprometida porque el procesador se encuentra trabajando a tiempo completo, por eso, es recomendable utilizar un buen equipo junto con su configuración VoIP.
- ✓ El protocolo IP es independiente de la capa de enlace, permitiendo que los usuarios finales elijan el formato de enlace más adecuado a las restricciones de costo y localización. IP puede viajar sobre ATM, Ethernet, Frame Relay, ISDN o incluso mediante líneas analógicas.

CAPITULO III

PROTOCOLO SIP

Introducción

En la actualidad las aplicaciones de voz y video se están convirtiendo en elementos claves para la comunicación entre personas. Cuando se desea implementar tecnologías que permitan el manejo de este tipo de aplicaciones multimedia, VoIP resulta ser la herramienta más apropiada.

La tecnología VoIP requiere para su funcionamiento el uso de un protocolo encargado de gestionar los recursos involucrados en la comunicación como: establecer, modificar y cerrar sesiones multimedia. De entre los cuales podemos mencionar el protocolo H323, SIP, AIX2. Siendo SIP el más utilizado actualmente.

En este capítulo se presenta los aspectos mas importantes del protocolo de inicio de sesión SIP; su historia, estructura, aplicaciones y funcionamiento. Debido a que el funcionamiento de SIP es a través del intercambio de mensajes entre el cliente y servidor; se detallan los mensajes de solicitud y respuesta SIP al mismo tiempo las estructuras de dichos mensajes. Además se muestran los problemas que se deben superar para permitir la comunicación SIP en el entorno NAT.

3.0 Protocolo SIP

El protocolo de inicio de sesiones (SIP, Session Initiation Protocol) desarrollado por el IETF, es un protocolo de señalización de capa de aplicación que define la iniciación, modificación y la terminación de sesiones interactivas de comunicación multimedia entre usuarios que incluyen conferencias y telefonía IP

El propósito de SIP es la comunicación entre dispositivos multimedia. SIP hace posible esta comunicación gracias a dos protocolos que son RTP y SDP. El protocolo RTP se usa para transportar los datos de voz en tiempo real (igual que para el protocolo H323, mientras que el protocolo SDP se usa para la negociación de las capacidades de los participantes, tipo de codificación, etc.)

3.1 Historia de SIP

El protocolo de inicio de sesión SIP fue desarrollado por el grupo MMUSIC (Multimedia Session Control) del IETF, definiendo una arquitectura de señalización y control para VoIP. Inicialmente fue publicado en febrero del 1996 en la RFC 2543, luego significativos cambios fueron hechos para el protocolo y resulto la versión 2.0, la cual fue presentada como un internacional borrador en 1998. El protocolo alcanzo el estatus Proyecto de Norma en marzo de 1999 y fue publicado como RFC 2543 en Abril de 1999. En septiembre de 1999, el grupo de trabajo de SIP fue establecido por la IETF para conocer el crecimiento interesante del protocolo. Un borrador de internet que contiene corrección de errores y aclaraciones para SIP fue proporcionado a comienzos de julio del 2000, refiriéndose como RFC 2543. Este documento fue eventualmente publicado como RFC3261 el cual remplazo al original RFC 2543.

3.2 Funcionalidades de SIP

3.2.1 Establecimiento, Modificación y Terminación de Sesión.

El estándar SIP define principalmente la forma en la que se lleva a cabo el establecimiento, modificación y terminación de comunicación multimedia. Es utilizado principalmente para invitar a nuevos usuarios a una sesión previamente establecida, o si se desea, permite establecer nuevas sesiones de comunicación con otros usuarios.

Para que el proceso de comunicación pueda llevarse a cabo, se requiere principalmente de dos aspectos, primero; que el usuario a invitar acepte participar en la sesión, y segundo, que estos usuarios establezcan los parámetros multimedia a utilizar.

Independientemente del tipo de sesión multimedia a utilizar, el funcionamiento de SIP es el mismo, al inicio de la comunicación, los usuarios establecen los códec de voz y video a utilizar, además de otros parámetros, hasta que esta parte no haya finalizo, no se puede iniciar la comunicación. Si durante la comunicación existen cambios como la desactivación del video por parte de uno de los participantes, se envía un mensaje para notificar a la otra parte del cambio realizado, y se mantienen los servicios que aun no se han desactivado; al momento que uno de los participantes decide terminar la comunicación, antes de llevarse a cabo la desconexión, se notifica a los interesados la finalización de la sesión.

3.2.2 Movilidad del usuario.

Una de las mayores ventajas de la telefonía IP es la posibilidad de contar con el servicio sin la necesidad de estar dentro de una red específica o utilizar en todos los casos el mismo equipo terminal. EL protocolo SIP antes de establecer la comunicación entre usuarios requiere que se conozca la dirección IP que poseen en

el momento los participantes de la sesión, para poder obtener dicha dirección el protocolo hace uso de las siguientes herramientas:

URLs SIP

En el caso de SIP una URL (localizador uniforme de recurso) se le asigna a cada usuario de la red con el objetivo de brindarle una referencia única en internet. Dicha URL esta formada por diferentes campos de información. De forma básica una URL SIP posee el siguiente formato:

SIP: // <user>: <password> @ <host>, <ttl>: <PORT>

El campo password es necesario en el URL ya que este permite la autenticación, si este campo falta en el URL, este genera un error de autenticación, por lo cual siempre debe de llenarse.

Las partes ": <password>", ", <ttl>", ": <PORT>" son opcionales. Si el puerto se omite, entonces el puerto TBD es asumido por el esquema SIP:

sipu:, y siprt: y el puerto TBD para el esquema sips:

El nombre de esquema URL también designa el protocolo de transporte que se utiliza por SIP. Para SIP: pueden ser usados ya sea UDP o TCP; según la preferencia de la entidad que resuelve el URL. sipu: y siprt:

Las solicitudes explícitamente usan de UDP Y TCP. sips: solicita el uso de SSL a través de TCP.

El valor TTL⁶ designa el valor de tiempo de vida de multicast. Este campo no debe ser utilizado si la parte del host de la URL es una dirección unicast. El valor por defecto es uno. El valor TTL es obligatorio si la dirección del host designa una dirección IPv6 multicast. (La representación textual del direccionamiento IPv6 contiene puntos y así no puede ser delimitada desde el puerto identificador.

⁶ (Time To Live - Tiempo de Vida). Contador en el interior de los paquetes multicast que determinan su propagación. Es un campo dentro del protocolo IP que especifica cuántos hops (saltos) puede dar un paquete antes de ser descartado o devuelto.

Algunos ejemplos de URL SIP son los siguientes:

- *sip:company.com*
- *sip:Usuario1@company.com*
- *sip:bob@192.168.1.1*
- *sip:bob:secret@company.com:5060*

Registro

Esta función de SIP permite a un usuario cambiar su ubicación (dirección IP). Al iniciar sesión se envía un mensaje SIP al servidor de registro donde se especifica el nombre de usuario SIP y la nueva dirección IP asociada a él. En caso que se desee establecer comunicación con este usuario el primer mensaje será dirigido al servidor de localización; la respuesta de este servidor puede ser de dos maneras: Redirección y Proxy. En el caso de redirección la respuesta al solicitante será la ubicación del destinatario, en caso de ser una respuesta Proxy, será el servidor el encargado de transferir la información de comunicación al destinatario.

3.3 Elementos SIP

3.3.1 Agentes De Usuario (UA)

Un propósito de SIP es activar sesiones que son establecidas entre los UA. Un agente de usuario es la entidad de SIP que interactúa con el participante de la comunicación. Esta usualmente tiene una interfaz hacia el usuario.

AUs son aplicaciones de punto final que envían y reciben peticiones SIP; dependiendo de su utilización pueden ser clientes de agentes del usuario (UAC) o servidores de agente del usuario (UAS). El UAC es el encargado de generar las invitaciones mientras que el UAS es el encargado de generar las respuestas. Durante una sesión, un user agent operará de ambas formas. En una sesión SIP, es obligatorio que los user agents soporten SDP ya que esto permitirá obtener una descripción del medio.

3.3.2 Servidor SIP

El servidor SIP son aplicaciones que aceptan peticiones SIP y responden a ellos. Un servidor SIP no podría confundirse con un agente servidor o cliente-servidor natural de protocolo, cual describe operaciones en términos de clientes (originador de las peticiones) y el servidor (originador de responde a las peticiones).

En resumen un servidor SIP es quien:

- Llevan a cabo funciones que pueden necesitar los puntos terminales.
-
- Típicamente actúa en respuesta a una petición de un terminal SIP.

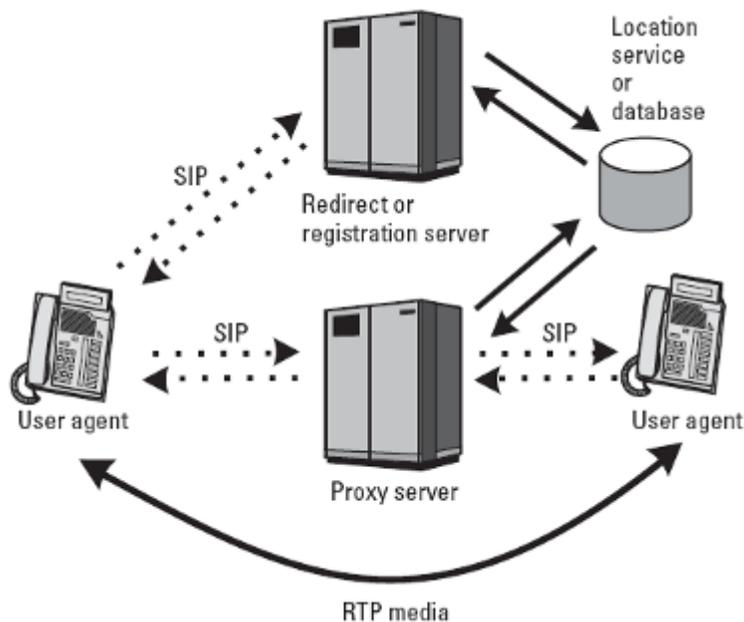


Figura 5: Interacción de los diferentes Servidores SIP y los Agentes de Usuarios.

3.3.2.1 Proxy SIP

Un servidor Proxy SIP recibe una petición SIP desde un agente de usuario o desde otro proxy y actúa en nombre del agente de usuario para la transmisión o para responder a la solicitud.

Un proxy puede usar cualquier número de tipos de base de datos para ayudar al procesamiento de una petición. La base de datos podría contener registros de SIP, presencia de información o cualquier otro tipo de información acerca de donde el usuario se encuentre.

No es necesario un servidor proxy para comprender una petición SIP, ya que su función es únicamente reenviarlas. Un proxy no debería cambiar el orden de los campos de cabecera o en general, modificar los o eliminar los campos de cabecera.

Un servidor Proxy es diferente desde un agente de usuario o un Gateway en tres claves:

1. Un servidor proxy no cuestiona las peticiones, este solo responde a ellas desde un agente de usuario.
2. Un servidor proxy no tiene la capacidad de los medios de comunicación.
3. Un servidor proxy no analiza el cuerpo del mensaje; si no que se basa exclusivamente en los campos de cabecera.

Tipos de Servidores Proxy

Los servidores proxy pueden ser clasificados de acuerdo a la cantidad de estados de información que ellos almacenan durante una sesión.

Un servidor proxy puede ser definido por tres tipos: Call stateful, stateful y stateless.

➤ Proxy Call Stateful

Este proxy necesita ser informado de todas las transacciones SIP que ocurren durante la sesión y por lo tanto, ellos siempre están en el camino tomados por los mensajes SIP viajando entre los usuarios finales. Estos servidores proxy guardan estados de información desde el momento que la sesión es establecida hasta el momento en que finaliza.

Un ejemplo del proxy call stateful es un servidor que implementa un servicio de llamada, tal como recibir un e-mail de conclusión en cada llamada con información acerca de la duración de la llamada.

➤ **Proxy Stateless.**

Un servidor proxy stateless procesa cada petición SIP o respuesta SIP basando solamente en el contenido de los mensajes.

Una vez el mensaje ha sido analizado, procesado y respondido, este no guarda información acerca del mensaje. Un servidor proxy stateless nunca res-tramite un mensaje, y este no usa ningún temporizador SIP.

➤ **Proxy Statefull.**

Un servidor proxy stateful mantiene un registro de las solicitudes y las respuestas recibidas en el pasado y utiliza esa información en la tramitación de las peticiones y respuestas. Por ejemplo, un servidor proxy stateful inicia un temporizador cuando una petición es solucionada. Si no responde a la petición esta no es recibida dentro del período de temporizador. El proxy podría retransmitir la petición, aliviando al agente de usuario de esta tarea.

El tipo más común de servidor proxy SIP es una transacción stateful proxy. Una transacción stateful proxy, mantiene estados sobre una transacción, pero solo para la duración que la petición esta pendiente.

Agentes de usuario y Servidor Proxy.

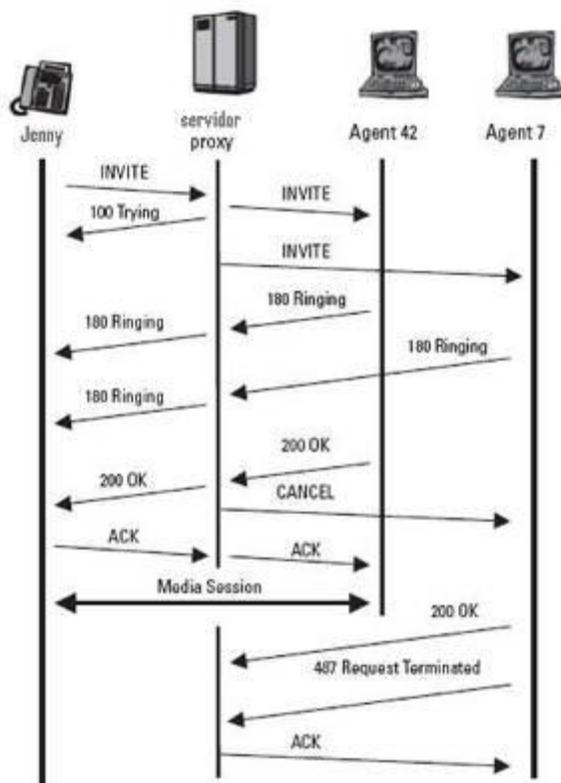


Figura 6: Interacción de los AU y el servidor Proxy SIP

3.3.2.2 Servidor de Localización

Proporciona Información acerca de la localización del usuario. Si un usuario desea comunicarse con otro usuario, en primer lugar el usuario que desea comunicarse, necesita descubrir la localización actual del usuario con el que se quiere comunicar en la red, con el fin de que la petición de establecimiento de sesión pueda llegarle. Además, hay que tener en cuenta que el usuario destino puede estar en diferentes lugares y en momentos distintos, incluso ser alcanzable por varios medios simultáneamente; por ejemplo, a través de una PC o un teléfono convencional.

3.3.2.3 Servidor de Redireccionamiento SIP

Acepta una petición SIP, mapea la dirección en cero o más direcciones nuevas y devuelve estas direcciones al cliente, a diferencia de los servidores proxy, los

servidores de redirección no inician su propia petición SIP. Tampoco pueden aceptar o terminar llamadas como ocurre con los UAS⁷

Agentes de Usuario y Servidor Proxy.

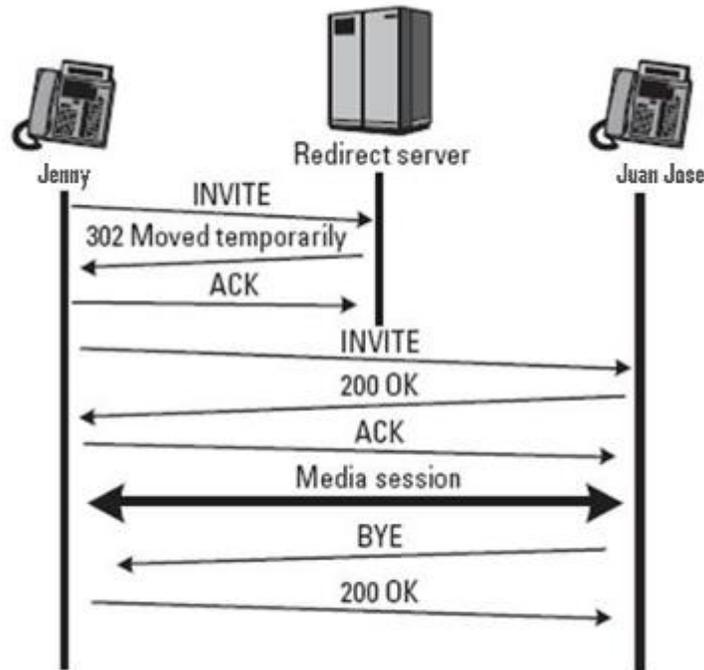


Figura 7: Interacción de los AU y el servidor Proxy de Redireccionamiento.

El servidor de redireccionamiento procesa los mensajes INVITE, que son solicitudes SIP, y retornan la dirección (o direcciones).

3.3.2.4 Servidor de Registro

El protocolo SIP permite establecer la ubicación física de un usuario determinado, esto es en qué punto de la red está conectado. Para ello se vale del mecanismo de registración.

⁷ UAS: USER Agent Server, agente de usuario de servidor: responsable de aceptar las peticiones de establecimiento de sesiones recibidas

El servidor de registro es el que realiza la asociación de la dirección física con la dirección lógica del usuario. Esta asociación tiene un período de vigencia y si no es renovada esta caduca.

Una dirección lógica del protocolo SIP es de la forma *usuario@dominio* es decir tiene la misma forma que una dirección de correo electrónico. La dirección física (denominada "dirección de contacto") es dependiente del lugar en donde el usuario está conectado (de su dirección IP).

3.4 Operación de SIP

3.4.1 Mensajes SIP.

El tercer elemento en una red VoIP corresponden a los mensajes SIP. Estos son en texto plano y emplean el formato de mensaje genérico establecido en la RFC 2822, es decir: Una línea de inicio, Campos de cabecera (header), Una línea vacía (indica el final del campo de cabeceras), Cuerpo de mensaje (opcional).

Los dos tipos de mensajes principales que se definen son de petición y de respuesta, cada mensaje contiene una cabecera que describe los detalles de la comunicación como rutas, tipo de mensajes, identificación del receptor y emisor, etc. En total SIP define 37 tipos de cabeceras.

Los mensajes se envían sobre TCP o UDP y pueden enviarse múltiples mensajes en un solo datagrama UDP o conexión TCP.

Línea de Inicio
Campo de Cabecera
Línea Vacía
Cuerpo del Mensaje

Tabla 8: Partes del Mensaje SIP

3.4.1.1 Mensaje de Solicitud.

Un mensaje de solicitud tiene distintas funciones, el propósito de una petición o solicitud, esta determinado por lo que se denomina Método de dicha petición; que no es más que un identificador del propósito de la petición.

- **MÉTODO**

La solicitud SIP o métodos son considerados “verbs” en el protocolo, desde ellos solicitan una específica acción para ser tomadas por otro agentes de usuario o servidor. El INVITE, REGISTER, BYE, ACK, CANCEL y OPTIONS son los seis métodos originales de SIP. El REFER, SUBSCRIBE, NOTIFY, MESSAGE, UPDATE, INFO y PRACK son métodos que se describen en RFC.

Hay que tener en cuenta que un proxy con el fin que se transmita la solicitud, un proxy trata aun método desconocido como si este fuera un OPTIONS; es decir, que remite la solicitud al destino si se puede. Esto permite nuevos rasgos y prácticos métodos para que los agentes de usuarios estén introducidos.

- **INVITE**

Cuando un usuario cliente desea iniciar una sesión (por ejemplo, audio, vídeo, o un juego), formula una petición INVITE. La petición INVITE pide establecer una sesión a un servidor. Esta petición puede ser enviada por los SIP Proxies, llegando eventualmente a unos o más SIP UAS que puedan potencialmente aceptar la invitación. Estos UASs necesitarán con frecuencia preguntar al usuario si acepta la invitación.

Un INVITE usualmente tiene un cuerpo de mensaje que contiene los medios de información de los usuarios que llaman. El cuerpo del mensaje puede contener también otras sesiones de información tales como Calidad de servicios (QoS) o seguridad de información. Si un Invite no contiene información en el medio, el ACK contiene la información del medio de la UAC.

Una sesión es establecida cuando el INVITE, 200 OK, y el mensaje ACK han sido intercambiados entre el Agente de Usuario cliente y el Agente de Usuario Servidor.

- **REGISTER**

El proceso de registrar exige enviar una petición REGISTER aun tipo especial de Agente de Usuario Servidor conocido como Registrar. Un SIP Registrar Server actúa como servicio de localización para los dominios, leyendo y escribiendo los mapeos basados en el contenido de las peticiones REGISTER. Este servicio de localización entonces es consultado típicamente por un proxy server que sea responsable de los pedidos de enrutamiento para ese dominio.

El método REGISTER es usado por un agente de usuario para notificar a la red SIP de estos actuales Contact URI⁸ (dirección IP) y el URI tendría rutas de peticiones para este Contact.

Los campos de cabecera que deben contener un mensaje REGISTER obligatoriamente son:

Call-ID, CSeq, From, To, Via y Max-Forwards

Un mensaje REGISTER puede contener un campo de cabecera llamado Expires que indica el tiempo por el cual el Contact será valido, también puede contener parámetros Expires que realizan la misma función. Si en el mensaje no se encuentra el campo Expires o e parámetro Expires el contac será valido por una hora.

Una petición REGISTER puede contener un cuerpo de mensaje, aunque su uso no esta definido en el estándar. Dependiendo de la utilización del Contac y Expires en las peticiones REGISTER, el registrar Server tomaría diferentes acciones.

- **BYE**

Las peticiones BYE son usadas para abandonar sesiones. En dos sesiones el abandono de una de las partes implica que la sesión se termina.

⁸ URI Uniform Resource Identifier: es un identificador más completo de las direcciones web, definido en RFC 2396.

Una sesión es considerada establecida si un INVITE ha recibido una exitosa respuesta o un ACK ha sido enviado. Un BYE es enviado solo por un agente que esta participando en la sesión, y nunca por proxies u otras terceras partes.

No es recomendable que un BYE sea usado para cancelar INVITEs pendientes, porque este no podría ser tratado como un INVITE y puede que no alcance los mismos grupos de agentes de usuario como el INVITE.

Los campos de cabecera que deben contener un mensaje de BYE obligatorio son: Call-ID, CSeq, From, To, Via y Max-Forwards.

- **ACK**

El método ACK es utilizado para confirmar la respuesta final a una petición INVITE. Así, un cliente origina una petición INVITE y se le emite una solicitud ACK cuando esta recibe una respuesta final para el INVITE. Las respuestas finales son definidas como 2xx, 3xx, 4xx, 5xx o 6xx.

Un ACK puede contener un cuerpo de mensaje Aplicación/SDP. Si el INVITE contiene un cuerpo de mensaje, el ACK puede que no contenga un cuerpo de mensaje.

El ACK no puede ser usado para modificar una descripción que ya han sido enviada en el inicial INVITE.

Los campos de cabecera que deben contener mensajes de ACK obligatorios son: Call-ID, CSeq, From, To, Via y Max-Forward.

- **CANCEL**

La petición CANCEL, como el nombre implica, se utiliza para cancelar una petición anterior enviada por un cliente. Específicamente, pide al agente de usuario servidor (UAS) que deje de procesar la petición y que genere una respuesta de error a esa

petición. El CANCEL no tiene ningún efecto sobre peticiones a las cuales un UAS ha dado ya una respuesta final.

El mensaje CANCEL puede ser generado por un agente de usuario o un Proxy cuando ante una petición solo se ha recibido una respuesta temporal 1xx y no una respuesta final. Cuando una respuesta final ha sido recibida no se puede utilizar un mensaje CANCEL para termina la sesión. El método Cancel es de tipo salto a salto.

Los campos de cabecera que debe contener un mensaje CANCEL obligatorios son: Call-ID, CSeq, From, To, Via y Max-Forwards.

- **OPTIONS**

El método OPTIONS es usado para preguntar al agente de usuario o al servidor acerca de sus capacidades y descubrir su disponibilidad actual, incluyendo cual método y cual sesión describe el soporte del protocolo. La respuesta para la petición, enlista las capacidades de un agente de usuario o servidor. Un proxy nunca genera una respuesta OPTIONS.

Una petición OPTIONS puede ser que no contenga un cuerpo de mensaje. Un proxy determina si una petición OPTIONS es para si misma mediante la exanimación de la petición URI. Si la petición URI contiene la dirección del proxy, la petición es para el proxy.

Los campos de cabecera que debe contener un mensaje OPTIONS obligatoriamente son: Call-ID, Cseq, From, To, Via y Max-Forwards.

De los métodos que se describen en RFC; mencionamos:

- **REFER**

El método REFER es utilizado por un usuario de agente para solicitar a otro agente de usuario el acceso a otro URI o URL. El URI o URL al que se desea acceder debe ser especificado en el campo de cabecera llamado Refer-To, el cual es un campo obligatorio en un mensaje REFER.

Una petición de REFER puede ser originada dentro o fuera de un dialogo existente y este puede ser utilizado para la transferencia de llamadas. Por ejemplo si una persona A esta en una llamada con otra persona denominada B, y esta (B) decide que necesita hablar también con la persona C, A puede hacer que su agente de usuario SIP envíe un petición REFER al agente de usuario de B proporcionando la información de contacto de la persona C. el agente de usuario de B intentara llamar a C usando ese contacto, el AU de B le hará entonces saber si tuvo éxito en lograr el contacto con C al AU de A.

Los campos de cabecera que debe contener un mensaje REFER obligatoriamente son: Call-ID, Cseq, From, To, Via, Contact, Refer-To y Max-Forwards.

- **SUCRIBE**

El método SUCRIBE es utilizado por un agente de usuario para establecer una suscripción que tiene como propósito, recibir notificaciones a través del método NOTIFY acerca de un evento particular. Una suscripción exitosa establece un dialogo entre el agente de usuario cliente y el agente de usuario servidor.

Cuando una petición SUBSCRIBE se envía a un servidor PINT⁹, indica que un usuario desea recibir información sobre el estado de una sesión de servicio. La petición identifica la sesión de interés al incluir la descripción original de la sesión junto con la petición, usando el identificador-de-sesión-global de SDP que forma parte del campo-origen para identificar únicamente la sesión de servicio.

La petición SUBSCRIBE se envía al mismo servidor que fue enviada el INVITE original, o a un servidor que fue especificado en el campo contacto dentro de una respuesta subsecuente (este bien podría ser el gateway PINT para la sesión).

Un mensaje SUSCRIBE posee un campo de cabecera llamado Expires que indica el tiempo de duración en la que la suscripción será valida. El tipo de evento acerca del cual se desea recibir notificaciones es especificado en el campo de cabecera

⁹ El servicio PINT define un protocolo para invocar ciertos servicios telefónicos de una red IP

llamado Event. El campo de cabecera llamado Allow-Events, un agente de usuario define el tipo de eventos que puede ser elegido.

Los tipos de eventos que pueden ser elegidos dentro de un método SUSCRIBE son: Conference, dialog, message-summary, presence, refer, reg, winfo.

Los campos de cabecera que debe contener un mensaje SUSCRIBE obligatoriamente son: Call-ID, Cseq, From, To, Via, Contact, Refer-To, Events y Max-Forwards.

- **NOTIFY**

El método NOTIFY es usado por un agente de usuario para transmitir información acerca de los acontecimientos de un particular evento. Un NOTIFY es siempre enviado con un dialogo cuando una suscripción existe entre el suscriptor y el notificador.

Una petición NOTIFY normalmente recibe 200 OK, para indicar que se ha recibido. Si un 481 DIALOG/TRASACTION No existe se recibe una respuesta, la suscripción es automáticamente terminada y no son enviadas más NOTIFY.

Un mensaje NOTIFY posee un campo de cabecera llamado Event, el cual identifica el evento en particular acerca del cual se esta informando, además contiene otro campo de cabecera llamado Subscriptio-State en el cual se informa acerca del estado de la suscripción.

La petición NOTIFY contiene la descripción modificada de la sesión.

Los campos de cabecera que debe contener un mensaje NOTIFY obligatorios son: Call-ID, Cseq, From, To, Via, Contact, Refer-To, Events y Allow-Events, Subscription-State y Max-Forwards.

- **MESSAGE**

El método MESSAGE, es una extensión al protocolo SIP que permite la transferencia de mensajes instantáneos (IM). Un IM usualmente consiste en un mensaje corto intercambiado en un tiempo real por los participantes en la conversación.

Puesto que la petición MESSAGE es una extensión al SIP, hereda todas las características de enrutamiento y de seguridad de ese protocolo. Las peticiones MESSAGE llevan el contenido en la forma de piezas de cuerpo MIME. Las peticiones MESSAGE no inician un diálogo SIP por si mismas; bajo uso normal cada mensaje instantáneo es independiente. Las peticiones MESSAGE se pueden enviar dentro del contexto de un diálogo iniciado por otra petición SIP.

El tamaño de las peticiones MESSAGE NO DEBE exceder los 1300 bytes, a menos que el UAC tenga conocimiento positivo de que el mensaje no atravesará un link congestionado-inseguro en alguno de los saltos, o que el tamaño del mensaje es por lo menos 200 bytes menor que el valor más bajo de MTU encontrado en el camino al UAS. Tamaños más grandes pueden ser enviados como parte de una sesión de medios, o de usar un cierto tipo de contenido-engañoso.

Una petición MESSAGE normalmente recibe una respuesta 200 OK que indican que el mensaje ha sido entregado a su destino final.

Los campos de cabecera que debe contener un mensaje MESSAGE obligatorios son: Call-ID, Cseq, From, To, Via, Max-Forwards.

- **INFO**

La función del método INFO es la de transportar la información de control que se genera durante una sesión. Un ejemplo de tal información de control de sesión son los mensajes de señalización de ISUP¹⁰ y de ISDN usados para controlar servicios de llamadas telefónicas.

Un mensaje de INFO típicamente posee un cuerpo de mensaje que puede contener información de señalización, eventos de la llamada o algún tipo de estímulo.

¹⁰ *ISUP* es un protocolo de circuitos conmutados, usado para configurar, manejar y gestionar llamadas de voz y datos sobre PSTN.

Los campos de cabecera que debe contener un mensaje INFO obligatorios son: Call-ID, Cseq, From, To, Via, Max-Forwards.

- **PRACK**

El método PRACK es utilizado para confirmar el recibido del mensaje de respuesta provisional, que son transportados de forma confiable.

La petición PRACK desempeña el mismo papel que el ACK , pero para las respuestas provisionales. Hay una diferencia importante, sin embargo PRACK es un mensaje normal de SIP, como el BYE. Como tal, su propia confiabilidad se asegura salto-a-salto con cada proxy. También como el BYE, PRACK tiene su propia respuesta, a diferencia del ACK. Si esto no fuese así, el mensaje de PRACK no podría atravesar los servidores proxy que cumplen con el RFC 2543.

Un PRACK es generado por un agente de usuario cliente, cuando provisionalmente la respuesta que ha sido recibida contiene un RSeq fiable de secuencias de números.

Los campos de cabecera que debe contener un mensaje PRACK obligatorios son: Call-ID, Cseq, From, To, Via, Rack y Max-Forwards.

- **UPDATE**

El mensaje UPDATE permite que un cliente actualice los parámetros de una sesión (tales como el set de media streams y de sus codecs) pero no tiene ningún impacto en el estado de un diálogo. En ese sentido, es como un Re-INVITE, pero a diferencia del re-Invite, este puede ser enviado antes de que se haya terminado el INVITE inicial. Esto lo hace muy útil para actualizar parámetros de la sesión.

El método UPDATE puede ser utilizado para realizar funciones como silencio, ejecutar QoS u otra negociación de atributos de extremo a extremo antes del establecimiento de la sesión.

Los campos de cabecera que debe contener un mensaje UPDATE obligatorios son: Call-ID, Cseq, From, To, Via, Contact y Max-Forwards.

SIP Y SIPS URIS.

Un SIP URI es el esquema de direccionamiento SIP para llamar a otra persona vía SIP. En otras palabras, un SIP URI es un número telefónico SIP de un usuario. El SIP URI es parecido a una dirección de correo electrónico y es escrito en el siguiente formato:

SIP URI = sip:x@y:Puerto

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

Ejemplos:

sip:joe.bloggs@212.123.1.213

sip:support@phonesystem.3cx.com

sip:22444032@phonesystem.3cx.com

El estándar SIP URI ha sido definido en RFC 3261.

Los SIP URIs son usados en un número de lugares incluyendo la cabecera del To, From, y el Contac, así como la petición-URI, cual indica la destinación. Los SIP URIs son similares al mailto URL¹¹ y pueden ser usados en un hiperlink sobre las páginas Web.

Algunos SIP URIs, tales como un REGISTER petición-URI no tiene un nombre de usuario, pero comienza con el host o con la dirección IPv4.

Un ejemplo de un código de respuesta:

INVITE sip:mark.rich@francetelecom.com SIP/2.0

Via : SIP/2.0/UDP station1.francetelecom.com:5060

Max-Forwards : 20

¹¹URLs mailto: es un esquema que se usa para designar una dirección de correo de Internet de un individuo o servicio

To : Mark Rich <sip:mark.rich@francetelecom.com>
From : Mary Taylor <sip:mary.taylor@francetelecom.com>
Call-Id: 23456789@station1.francetelecom.com
CSeq: 1 INVITE
Contact: mary.taylor@192.190.132.20
Content-Type: application/sdp
Content-Length: 162

Cuando el destinatario acepta la sesión, la respuesta 200 OK esta emitida por su UA y encaminada hacia la UA del que genera la llamada.

SIP/2.0 200 OK

Via : SIP/2.0/UDP ps1.francetelecom.com:5060
Via : SIP/2.0/UDP station1.francetelecom.com:5060
Max-Forwards : 20
To : Mark Rich <sip:mark.rich@francetelecom.com>
From : Mary Taylor <sip:mary.taylor@francetelecom.com>
Call-Id: 23456789@station1.francetelecom.com
CSeq: 1 INVITE
Contact: mark.rich@192.190.132.27
Content-Type: application/sdp
Content-Length:162

3.4.1.2 Mensaje De Respuesta

Cada petición recibida por parte de un User agent Client (UAC), genera de manera automática uno o mas mensajes de repuesta por parte de User Agent Server (UAS). El objetivo de estos es indicar al emisor el estado de la transacción, dichos mensajes pueden ser de seis tipos:

1XX Mensajes Informativo (Informational)

Este tipo de respuesta posee el código 1xx; su principal función es indicar el estado de la llamada antes de ser completada, es información de punto a punto, es

considerada una respuesta provisional, antes de enviar una respuesta final del tipo 2xx, 3xx 4xx, 5xx o 6xx. De esta manera se evita la retransmisión de mensajes **Invite** por parte del UAC. Una respuesta **informativa** puede ser omitida durante el proceso de comunicación entre el cliente y el servidor, si se envía un tipo de mensaje con código superior al 1xx.

- **100 Tratando (Trying)**

Este mensaje puede ser generado por parte de un servidor proxy o un agente de usuario, simplemente indica que se esta realizando alguna acción o proceso que tomara un periodo de tiempo; por ejemplo en el caso de un proxy puede indicar que esta en el proceso de búsqueda de la ubicación de un usuario. No posee cuerpo de mensaje y no genera ningún mensaje de respuesta adicional.

- **180 Teléfono sonando (Ringing)**

Esta respuesta indica que la petición Invite ha sido recibida por el agente usuario, y que se esta produciendo una alerta en el lado del receptor. El cuerpo del mensaje puede transportar información de seguridad o aplicación de calidad de servicio (QoS).

- **181 Redirección de llamada (Call Is Being Forwarded)**

La respuesta indica que la llamada ha sido trasferida a otro dispositivo terminal. Este mensaje es enviado cuando la información pueda ser utilizada por el emisor de la comunicación, debido a que al momento de realizar esta operación, pueda llevar mas tiempo a la llamada ser aceptada por el receptor.

- **182 Llamada puesta en cola (Call Queued)**

Se produce cuando un mensaje **invite** ha sido recibido satisfactoriamente, pero esta solicitud de comunicación ha sido puesta en cola pendiente de ser aceptada por el dispositivo final. Se puede utilizar este mensaje para estimar el tiempo de espera o el número de llamadas que se encuentran pendientes, al mismo tiempo el cuerpo del mensaje puede transportar música de espera o algún otro tipo de comunicación multimedia.

- **183 Progreso de Sesión (183 Session Progress)**

Se utiliza para transportar información del progreso de la llamada que no está clasificada de otra manera. Dicha información puede estar presente en el texto asociado a la respuesta, en los campos cabecera, en el cuerpo del mensaje o en el flujo de información del medio. A diferencia de la respuesta 100 Trying, la respuesta 183 establece una comunicación punto a punto entre los dispositivos, al tampoco transporta información específica del estado de la solicitud Invite.

2XX Mensajes de éxito (Success)

Este tipo de respuestas poseen un código 2xx e indican que la petición de comunicación ha sido aceptada.

- **200 OK**

Esta respuesta tiene dos usos en SIP. El primero para aceptar una invitación de sesión (INVITE), en cuyo caso contendrá un cuerpo de mensaje con las propiedades del medio del UAS (parte llamada). El segundo como respuesta a otras peticiones, indicando que la petición se ha recibido con éxito. Contiene un cuerpo de mensaje al recibir solicitudes Option y Register. En cambio para las solicitudes Info, Message, Subscriber, Notify, Prack; un cuerpo de mensaje no es permitido.

Esta respuesta detiene posteriores retransmisiones de futuras peticiones.

- **202 Aceptada (Accepted)**

Esta respuesta indica que el UAS ha recibido y comprendido la petición, pero que la petición puede no haber sido autorizada o procesada por el servidor. Es comúnmente utilizada como respuesta a solicitudes: Suscribe, refer y en algunos casos message.

3XX Mensajes de redirección (Redirection)

La respuesta es enviada por un servidor SIP que actúa como servidor redirect, en respuesta a una solicitud INVITE; dirigiendo al cliente a un contacto elegido entre un conjunto de direcciones URI alternativas; Asimismo, un UAS puede enviar una

respuesta de esta clase en el caso de que estén implementados los servicios de desvío de llamada.

- **300 Múltiples opciones** (*Multiple Choices*)

Esta respuesta de redirección contiene múltiples direcciones de contacto (campos Contact), las cuales indican que el servicio de localización ha devuelto diferentes localizaciones posibles para el Request-URI de la petición SIP. Se da, en caso que el usuario solicitado posea más de una ubicación posible.

- **301 Movido permanentemente** **Moved Permanently**

Esta respuesta se genera cuando un usuario cambia de ubicación, contiene un campo cabecera Contact que indica la nueva dirección URI de la parte llamada. El cliente que realiza la petición deberá actualizar su lista de direcciones con la nueva dirección para tenerla en cuenta en las siguientes peticiones.

- **302 Movido temporalmente** **Moved Temporarily**

La dirección URI incluida en esta respuesta tiene una validez temporal, por el tiempo indicado en la cabecera Expires o en el parámetro expires del campo Contact y por tanto dicha dirección puede ser guardada en el proxy o UAS para posteriores transacciones durante el tiempo indicado en dicho parámetro o campo. En caso de que no se indique explícitamente la duración de la validez de la citada dirección, ésta sólo será válida por una vez y por tanto no debe ser guardada.

- **305 Utiliza Proxy** (**Use Proxy**)

Esta respuesta contiene la dirección URI para dirigir las peticiones a un servidor proxy que tiene información autorizada sobre la parte llamante. Es decir, al recurso requerido debe accederse a través del servidor proxy. La dirección del proxy vendrá en el campo Contact de la respuesta y será a la que el cliente dirigirá de nuevo la petición.

- **380 Servicio alternativo** (**Alternative Service**)

Se produce en situaciones en las que no se ha podido completar la llamada pero existen servicios alternativos, como por ejemplo el desvío a un buzón de voz. Esta respuesta devuelve una dirección URI en función del tipo de servicio activado por la parte llamada.

4XX Mensajes de error por parte del Cliente (Client error)

Esta clase de respuesta es usada por un servidor o UAS para indicar que la petición no puede ser formulada tal y como se ha sido enviada. El tipo de respuesta o la presencia de determinados campos de cabecera, indicarán al UAC la naturaleza del error y cómo debe ser formulada de nuevo la petición.

- **400 Solicitud errónea (Bad Request)**

Esta respuesta indica que la petición no ha sido entendida por el servidor debido a algún error de sintaxis.

- **401 No Autorizado (Unauthorized)**

La respuesta indica que la petición requiere llevar a cabo el procedimiento de autenticación. Esta respuesta es generalmente enviada por un agente usuario.

Un ejemplo de este error es:

```
SIP/2.0 401 Unathorized
Via: SIP/2.0/UDP proxy.globe.org:5060;branch=z9hG4bK2311ff5d.1
;received=192.0.2.1
Via: SIP/2.0/UDP 173.23.43.1:5060;branch=z9hG4bK4545
From: <sip:explorer@geographic.org>;tag=341323
To: <sip:printer@maps-r-us.com>;tag=19424103
From: Copernicus <sip:copernicus@globe.org>;tag=34kdilsp3
Call-ID: 123456787@173.23.43.1
CSeq: 1 INVITE
WWW-Authenticate: Digest realm="globe.org",
nonce="8eff88df84f1cec4341ae6e5a359", qop="auth",
opaque="", stale=FALSE, algorithm=MD5
Content-Length: 0
```

- **402 Pago requerido (Payment Required)**

Esta respuesta se mantiene para un uso futuro. Podría ser utilizada para negociar cargos de complementación de llamadas.

- **403 Prohibido (Forbidden)**

Esta respuesta se utiliza para denegar una petición sin dar opción al llamante. En este caso el servidor ha entendido la petición y está correctamente formulada pero no atenderá la petición. Esta respuesta no es utilizada en caso que se requiera autorización.

- **404 No encontrado (Not Found)**

Esta respuesta se proporciona cuando el servidor tiene seguridad de que el usuario identificado por la dirección URI no existe en el dominio especificado en el *Request-URI*.

También se envía si el dominio no es ninguno de los dominios manejados por el receptor de la petición.

- **405 Método no permitido (Method Not Allowed)**

En este caso la solicitud especificada en el *Request-Line* ha sido comprendida correctamente por el servidor o agente de usuario pero no está permitido su uso para la dirección identificada en el *Request-URI*.

- **406 No Aceptable (Not Acceptable)**

El recurso identificado por la petición es únicamente capaz de responder con características de contenido no aceptables según el campo cabecera *Accept* incluido en la petición.

- **407 Autenticación Proxy Requerida (Proxy Authentication Required)**

Esta respuesta se envía desde un *proxy* para indicar al UAC que debe primero autenticarse antes de que la petición pueda ser procesada.

- **408 Expiración de solicitud (Request Timeout)**

Se genera cuando el campo de cabecera Expires está presente en una solicitud Invite, y el periodo de tiempo establecido en ese campo ha sido sobrepasado. Además puede ser enviada cuando la solicitud no genera alguna respuesta en el servidor proxy.

- **410 Ido (Gone)**

Es similar a la respuesta 404 pero proporciona información de que el usuario requerido no estará disponible en su posición en el futuro. El servidor utilizará esta respuesta cuando tenga seguridad de que se trata de una condición permanente, en caso de que no exista tal seguridad deberá emplear la respuesta 404.

- **415 Tipo de medio no soportado (Unsupported Media Type)**

Esta respuesta es enviada desde un Agente de usuario para indicar que el tipo de medio solicitado en una solicitud Invite no es soportada.

- **422 intervalo de tiempo de sesión demasiado corto (Session Timer Interval Too Small)**

Se usa para rechazar una petición que contiene un campo cabecera *Session-Expires* con un intervalo de tiempo demasiado corto. El intervalo de tiempo mínimo permitido es el indicado en el campo cabecera *Min-SE*.

- **423 Intervalo muy corto (Interval Too Brief)**

La usa un servidor de registro para rechazar una petición debido a que el tiempo en el que expira uno o más contactos (*Contact*) es demasiado corto.

- **480 Temporalmente no disponible (Temporarily Unavailable)**

Sirve para indicar que la petición ha alcanzado el destino correcto pero la parte llamada no está disponible por alguna razón (por ejemplo tiene activado el servicio “no molesten”). El texto asociado dará información más detallada de la causa por la que no está disponible. Además debe incluir un campo de cabecera *Retry-After* indicando el tiempo que puede esperar a enviar una nueva solicitud de conexión.

- **481 Llamada/Transacción no existe (Dialog/Transaction Does Not Exist)**

Indica que el UAS ha recibido una petición para la cual no encuentra una transacción o diálogo existente.

- **482 Lazo detectado (Loop Detected)**

Indica que la petición ha entrado en un bucle, ya que ha sido devuelta a un *proxy* que previamente transfirió la petición.

- **483 Muchos saltos (Too Many Hops)**

Indica que la petición ha sido desviada un número de veces que supera el máximo permitido. El servidor que manda esta respuesta ha recibido en la petición el campo cabecera *Max-Forwards* puesto a 0.

- **486 Ocupado Aquí (Busy Here)**

Se usa para indicar que, aunque se ha alcanzado correctamente a la parte llamada, el agente de usuario no puede aceptar la llamada en la posición cuya dirección se identifica en el *Request-URI*.

- **487 Solicitud Terminada (Request Terminated)**

Se enviará como respuesta a la cancelación de una solicitud Invite que se encuentra pendiente. O como respuesta un Bye.

- **488 No aceptable acá (Not Acceptable Here)**

Indica que el agente de usuario fue contactado correctamente pero que algunos aspectos de la descripción de la sesión, tales como el medio requerido, el ancho de banda o el esquema de direccionamiento no son aceptables.

- **491 Solicitud pendiente (Request Pending)**

Se usa para resolver posibles re-INVITEs simultáneos realizados por ambas partes del diálogo. Desde que ambos INVITE miran el cambio de estado de la sesión, ellos no pueden ser procesados en el mismo tiempo. Ejemplo

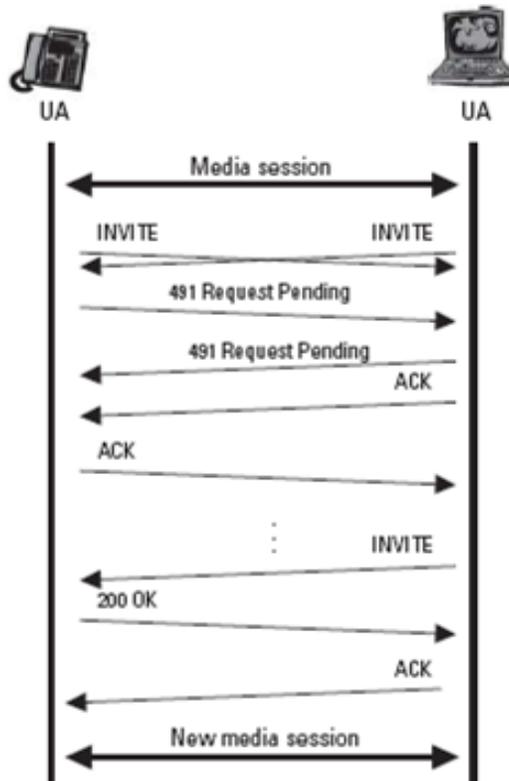


Figura 8: Ejemplo de Solicitud INVITE

- **493 No descifrable (Request Undecipherable).**

Esta respuesta es usada por el UAS cuando no puede descifrar el cuerpo de mensaje S/MIME al no disponer éste de la clave pública. Si el UAS no soporta S/MIME, no se presentara ningún cuerpo de mensaje.

5XX Mensajes de error por parte del Servidor (Server error)

Esta clase de respuestas se usará para indicar que la petición no se puede procesar debido a un fallo en el propio servidor. La petición podrá ser reintentada para otras direcciones.

- **500 Error Interno del Servidor (Server Internal Error)**

Esta respuesta se envía cuando el servidor se ha encontrado con un fallo inesperado que no le permite procesar la petición. Se trata de fallos temporales, por tanto, el cliente puede hacer un nuevo intento transcurridos unos segundos.

- **501 No Implementado (*Not implemented*)**

Indica que el servidor no es capaz de procesar la petición. Esta respuesta es apropiada cuando el UAS no reconoce el Método requerido. La diferencia con la respuesta 405 es que en esta última el servidor sí reconoce el método pero no es soportado o no está permitido.

- **502 Pasarela errónea (*Bad Gateway*)**

Esta respuesta se envía desde un *proxy* que está actuando como *Gateway* de otra red e indica que existe algún problema en la otra red que impide procesar la petición.

- **503 Servicio no disponible (*Service Unavailable*)**

Indica que el servicio requerido está temporalmente indisponible por congestión o actuaciones de mantenimiento del servidor. La solicitud puede ser reenviada después de unos segundos, o después que expire el tiempo dado en el campo de cabecera *Retry-After*.

- **504 Expiración de tiempo en la pasarela (*Gateway Timeout*)**

Esta respuesta indica que la petición ha fallado debido a un vencimiento de la temporización que se ha producido en el servidor o en la otra red con la que se interconecta el *Gateway*.

- **505 Version no soportada (*Version Not Supported*)**

Se produce esta respuesta cuando el servidor rechaza la petición debido a la versión SIP empleada en la petición.

- **513 Mensaje demasiado largo (*Message Too Large*)**

Esta respuesta es usada por el UAS para indicar que el tamaño de la petición es demasiado grande para ser procesado.

6XX Mensaje de Falla global (*Global failure*)

Esta clase de respuesta indica que el servidor sabe que la petición fallará en cualquier caso donde se intente. Como consecuencia, no debería reintentarse a otras direcciones.

- **600 Ocupado en Todas Partes (Busy Everywhere)**

Esta respuesta es la versión definitiva de la respuesta 486 Busy Here, es decir, tiene el mismo significado pero referido no sólo a una dirección sino a cualquier posible dirección del usuario identificado en el *Request-URI*.

- **603 Declinación (Decline)**

Es una respuesta similar a la 600 pero no proporciona información del estado de la llamada en el servidor. Puede indicar que la llamada no se acepta, porque no quiere o porque no puede.

- **604 No existe en ninguna parte (Does Not Exist Anywhere)**

Esta respuesta es similar a la 404 pero indica que el usuario identificado no puede ser localizado en ninguna dirección. Esta respuesta es enviada por un servidor que posea acceso a toda la información del usuario.

- **606 No Aceptable (Not Acceptable)**

Esta respuesta se podrá usar para implementar alguna capacidad de negociación de sesión en SIP. Sirve para indicar que algún aspecto de la sesión requerida no es aceptable por el UAS (medio requerido, ancho de banda, estructura de direccionamiento, etc.) y en consecuencia, no se puede establecer la citada sesión.

3.4.1.3 Cabecera de Mensaje SIP

Los campos de cabecera especifican las llamada, emisor de la llamada, la trayectoria del mensaje, tipo y largo del cuerpo del mensaje entre otras características.

El número total de cabeceras definidas en el protocolo SIP son 46, aunque en la definición inicial de SIP eran solo 37.

Cabeceras Generales	Cabeceras de Identidad	Cabeceras de Solicitud	Cabeceras de Respuesta
Call-ID	Allow	Accept	Proxy-Authenticate
Contact	Content-Encoding	Accept-Encoding	Server
Cseq	Content-Length	Accept-Languaje	Unsupported
Date	Content-Type	Accept-Contact	Warning
Encryption	Content-Disposition	Autorizacion	WWW-Athenticate
From	Expires	Hide	Rseq
Organization	MINE-Version	In-Reply-To	
Retry-After		Max-Forwads	
Subject		Priority	
Timestamp		Proxy-Authoration	
To		Proxy-Require	
User Agent		Record-Route	
Vía		Reject-Contact	
		Request-Disposition	
		Require	
		Response-Key	
		Route	
		Rack	
		Session-Expires	

Tabla 9: Los diferentes campos de los cuatro tipos de cabeceras del mensaje SIP

Los distintos tipos de cabeceras SIP se pueden dividir en cuatro tipos:

- *cabeceras generales*: aplicadas tanto a los mensajes de peticiones como a los de respuesta.
- *cabeceras de entidad*: definen información sobre el cuerpo del mensaje. Si el cuerpo no está presente, sobre los recursos identificados por la petición.
- *cabeceras de solicitud*: actúan como modificadores de solicitud. Permiten que el cliente pase información adicional sobre la solicitud o sobre si mismo.
- *cabeceras de respuesta*: permiten al servidor agregar información adicional sobre la respuesta cuando no hay lugar en la línea de inicio (Status-Line).

A continuación se muestran los cuatro grupos de cabeceras y los campos que las componen. Sin embargo los campos obligatorios son To, From, Vía, CSeq, Call-ID y Max-Forwards. Los cuales se describen a continuación.

Call-ID

El campo Call-ID es utilizado para identificar una llamada entre dos agentes de usuario. El Call-Id debe ser único entre todas las llamadas, excepto en el caso de peticiones de registro en las cuales se debe mantener el mismo Call-ID para todas las peticiones realizadas.

El Call-ID esta constituido de un identificador local, normalmente aleatorio o un simple identificador, el símbolo @ y el nombre del host o la dirección de IP. Debido a que un agente de usuario puede asegurar que su identificador local sea único, con la adición del nombre del host, el cual es único en toda la red; se puede asegurar que el Call-ID será globalmente único.

Las solicitudes SIP contienen algunas cabeceras SIP después de la línea de petición, mientras que las respuestas SIP ponen la cabecera después del estado de la línea.

CSeq

El campo de cabecera CSeq contiene un número decimal el cual es aumentado en cada petición realizada. El CSeq es utilizado por un agente de usuario servidor para determinar peticiones que se encuentra fuera de secuencia o para diferenciar entre una nueva petición y una retrasmisión de una petición anterior. Un agente de usuario cliente utiliza el CSeq para hacer corresponder una respuesta recibida con la petición a la que hace referencia.

Cada agente de usuario mantiene su propio espacio para su CSeq, el cual ira aumentando con cada petición realizada y no será afectado por las peticiones recibidas desde otro agente de usuario que poseen su propio espacio para su Cseq, el cual es totalmente independiente.

From

El campo de cabecera From indica quien es el autor de la petición.

El campo From contiene el URI del autor de la petición y además presenta un tag, el cual es utilizado para identificar una llamada en particular. En adición al campo From puede contener un nombre que sirva para identificar al autor de la petición, si este es el caso tanto el URI y el Tag estarán encerrados dentro de < >.

To.

El campo de cabecera To indica quien es el receptor de la petición. El campo To contiene el URI del rector de la petición y además presenta un Tag, el cual es agregado por el agente de usuario cuando genera una respuesta para la petición original. En adición el campo To puede contener un nombre que sirva para identificar al receptor de la petición, si este es el caso tanto el URI y el Tag estaría encerrados dentro de < >.

Vía

El campo Vía es utilizado para registrar la ruta SIP tomada por una petición y es utilizado para encaminar la respuesta de regreso al autor de la petición. Un agente

de usuario que genere una petición registra su dirección en un campo Vía en el mensaje de petición.

Este campo también contiene una etiqueta llamada Branch que consiste en una mezcla criptográfica de los campos To, From, Call-ID, CSeq y del Request-URI. Cuando el agente de usuario servidor genera su respuesta copia todos los campos Vía que se encontraban en el mensaje original y los copia en el mensaje de respuesta. Cuando un servidor proxy recibe el mensaje de respuesta a una petición, este revisa la lista de los campos Vía, si la primera dirección de la lista no es la suya descarta el mensaje debido a un mal encaminamiento, pero si la primera dirección de la lista coincide con su dirección el servidor remueve su campo Vía y reenvía el mensaje.

El campo Vía contiene el nombre del protocolo, el número de la versión y el nombre del protocolo de transporte.

Max-Forwards

El campo Max-Forwards indica el número máximo de saltos que un mensaje SIP puede realizar. El valor del campo se disminuye en uno cada vez que un proxy reenvía el mensaje. Cuando un servidor proxy recibe un mensaje cuyo campo de Max-Forwards posee un valor de cero, este descarta al mensaje y envía un mensaje 483 Too Many Hops al autor del mensaje.

En si los campos de cabeceras proveen información acerca de la petición (o respuesta) y acerca del cuerpo que estas contienen. Algunas cabeceras pueden ser usadas en ambas peticiones y repuestas, pero otras solo son específicamente peticiones o respuestas. Las cabeceras son un token que representa el nombre del campo de cabecera seguida por los dos puntos, y luego seguido por el valor de la cabecera.

Por ejemplo, el llamado de cabecera From, quien identifica el originador de una solicitud en particular, se mira como el siguiente.

From, Bob Johnson <sip,Bob.Johnson@company.com>

Como se puede observar en el ejemplo, un valor de cabecera puede tener varios campos. En el ejemplo, el From de cabecera tiene dos campos: un nombre de una persona y su URL SIP.

Ejemplo de la cabecera de los paquetes de Petición y de Respuesta.

Mensaje de Petición.

REGISTER sip:192.168.28.124:5060 SIP/2.0	Línea de Inicio
From: <sip:201@192.168.28.210> To: <sip:201@192.168.28.124> Call-ID: d0b54a5788484465af65835507b2e47f@192.168.28.210 CSeq: 1 REGISTER Via: SIP/2.0/UDP 192.168.28.210:1855 Max-Forwards: 2 Contact: <sip:201@192.168.28.210:1855;transport=udp> Content-Length: 0	Cabeceras
	Línea en Blanco
	Cuerpo del mensaje

Mensaje de Respuesta.

SIP/2.0 200 OK	Línea de Inicio
From: <sip:201@192.168.28.124> To: <sip:201@192.168.28.210> Call-ID: d0b54a5788484465af65835507b2e47f@192.168.28.210 CSeq: 1 REGISTER Via: SIP/2.0/UDP 192.168.28.210:1855 User-Agent: Asterisk PBX Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, SUBSCRIBE, NOTIFY Max-Forwards: 70 Expires: 120 Contact: <sip:201@192.168.28.210:1855;transport=udp>;expires=120 Date: wed, 18 oct 2006 20:24:17 GMT Content-Length: 0	Cabeceras
	Línea en Blanco

3.4.1.4 Cuerpo de Mensaje SIP

Ambos tipos de mensajes SIP Solicitud y Respuesta, pueden incluir un cuerpo de mensaje, separado del campo de cabecera por una línea en blanco.

Este cuerpo de mensaje se utiliza para transportar descripciones de las sesiones. Esta información es comunicada extremo a extremo por parte de los Agentes de Usuario (UA), es transparente a los proxy involucrados, debido a que los proxy SIP

únicamente necesitan conocer la información contenida en los campos de Estado y Solicitud de la cabecera para enrutar los mensajes.

En caso que deba utilizarse un cuerpo de mensaje, el contenido debe realizarse siguiendo el protocolo SDP, el cual define dieciséis parámetros de información, de los cuales cinco son obligatorios y los demás opcionales.

	Campo	Nombre	Utilización
1	v=	Versión del protocolo	Obligatorio
2	o=	Dueño/creador e identificador de sesión	Obligatorio
3	s=	Nombre de la sesión	Obligatorio
4	i=	Información de la sesión	Opcional
5	u=	Uniform Resource Identifier (URI)	Opcional
6	e=	Dirección email	Opcional
7	p=	Número de teléfono	Opcional
8	c=	Información de la conexión	Obligatorio
9	b=	Información de ancho de banda	Opcional
10	t=	Tiempo de la sesión activa	Obligatorio
11	r=	Tiempos de repetición	Opcional
12	z=	zona de tiempo	Opcional
13	k=	Llave del cifrado	Opcional
14	a=	Atributos de sesión	Opcional
15	m=	Nombre de los medios y dirección de transporte	Opcional
16	a=	Cero o más líneas de atributos de medios	Opcional

Tabla 10: Campos del cuerpo del mensaje SIP

Parámetros SDP.

Versión del protocolo.

El campo **v=** contiene el número de versión del protocolo SDP; debido a que la versión actual es la cero (0) un parámetro válido sería:

v=0

Dueño/creador e identificador de sesión.

Contiene la información del creador de la sesión y un identificador. Con este campo se logra dar a la sesión un identificador de sesión único. Los parámetros que posee son los siguientes:

o=username session-id version network-type address-type address

De los cuales:

username: contiene el nombre de usuario o host que inicia la sesión.

session-id: es un timestamp¹² del protocolo NTP(Network Time Protocol)¹³o un número aleatorio; de esta manera se proporciona un identificar único de sesión.

Versión: es un campo numérico que se incrementa con cada cambio que se da durante la sesión.

network-type: el valor de este campo será IN por tratarse de Internet.

address-type: puede tener los valores IP4 o IP6 dependiendo de la versión de dirección IP, seguido de dicha dirección o de un nombre de host valido.

Nombre de la sesión

El campo s= contiene un nombre de sesión diferente de cero.

Información de la sesión

Este campo i=, es opcional; y es utilizado para describir cualquier información adicional de la sesión.

Uniform Resource Identifier (URI)

El valor opcional de u= contiene el identificador URI, además de alguna otra información de la sesión.

Dirección E-Mail

La dirección e-mail de los participantes puede incluirse en el campo e=, además de un nombre de usuario.

Numero de teléfono.

Este valor se describe en el campo p=, dicho numero telefónico debe estar en formato globalizado, debe iniciar con un signo + seguido del código de país, espacio

¹² *Timestamp*: es la palabra utilizada para identificar la introducción de una fecha de creación o modificación en los archivos de sistemas informáticos.

¹³ *Network Time Protocol (NTP)*: es un protocolo de internet para sincronizar los relojes de los sistemas informáticos a través de ruteo de paquetes en redes con latencia variable.

o un signo -, luego el numero local. Este campo puede contener comentarios siempre que estén escritos entre paréntesis.

Información de la conexión.

El campo c= contiene información del medio de conexión contiene los siguientes valores:

C=tipo-de-red tipo-de-dirección dirección-de-conexión

tipo-de-red: este parámetro tiene el valor IN por tratarse de Internet.

tipo-de-dirección: se definen los valores IP4 para direcciones IP versión cuatro e IP6 para IP versión seis.

dirección-de-conexión: indica cual es la dirección a la que se dirigirán los paquetes multimedia.

Información de ancho de banda

El campo b= contiene información del ancho de banda requerido para la sesión, contiene los siguientes valores:

b=modificador:ancho-de-banda

El campo modificador especifica dos opciones: CT que se utiliza en sesiones multicast, para especificar cual es el valor de ancho de banda total que estará disponible para los participantes. Y el valor AS especifica el ancho de banda de un solo destino. Ancho de banda será el valor en kilobytes por segundo.

Tiempo de la sesión activa.

Este campo se utiliza para especificar el tiempo de inicio y finalización de la sesión.

t= start-time stop-time

Estos tiempos se especifican utilizando timestamps del protocolo NTP. Para una sesión programada un stop-time de cero indica que se mantendrá de manera indefinida. En cambio si ambos valores start-time y stop-time poseen valores de cero, significa que se trata de una sesión permanente.

Cero o más tiempos de repetición y zona de tiempo

Estos campos son opcionales y contienen información acerca del tiempo de repetición y la zona horaria.

Llave del cifrado.

Es un campo opcional que contiene la llave de cifrado a ser utilizada por los participantes de la sesión.

k=method:encryption-key

Nombre de los medios y dirección de transporte.

El campo opcional m= contiene información acerca del tipo de sesión multimedia. Este campo contiene:

m=media port transport format-list

El parámetro media puede ser audio, video, datos, eventos del teléfono, o control. El parámetro port es el número de puerto. Transport nos indica el protocolo de transporte y format-list contiene más información acerca de los datos multimedia.

Atributos.

Este campo a= es utilizado para extender las funcionalidades de SDP, y contiene atributos de nivel de sesión o a nivel de multimedia.

Un ejemplo del cuerpo de mensaje utilizando este protocolo es el siguiente:

```
v=0
o=johnston 2890844526 2890844526 IN IP4 43.32.1.5
s=SIP Tutorial
i=This broadcast will cover this new IETF protocol
u=http://www.digitalari.com/sip
e=Alan Johnston alan@mci.com
p=+1-314-555-3333 (Daytime Only)
c=IN IP4 225.45.3.56/236
b=CT:144
```

```
t=2877631875 2879633673
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000
m=video 23422 RTP/AVP 31
a=rtpmap:31 H261/90000
```

3.4.2 Autenticación.

Se utiliza un mecanismo de desafío-respuesta (challenge-response), basado en uno de los esquemas de autenticación (*Digest* y *Basic*) definidos en HTTP (RFC 2617). En concreto, en SIP se utiliza el esquema *Digest*, pues presenta sobre el esquema *Basic* la ventaja de que la password del usuario viaja codificada.

El procedimiento general de autenticación consiste en lo siguiente:

Si servidor de registro, UAS o proxy recibe una petición, para la que necesita disponer de autenticación, analiza la acreditación o credenciales incluidas en la misma. Si no las lleva o son incorrectas, envía al UAC una respuesta indicando que debe reenviar la petición con una acreditación correcta (en esto consiste el “challenge”).

Esta respuesta será distinta según el tipo de elemento SIP que solicite la autenticación, pero en cualquier caso, indicará el esquema de autenticación utilizado (en este caso “Digest”), y llevará información suficiente para que el UAC pueda proporcionar la acreditación adecuada. Dicha información incluye el espacio de protección (parámetro “realm”) dentro del cual podrá utilizarse esa acreditación con el correspondiente usuario-password, y un valor generado por el servidor que se utiliza para validar la acreditación (parámetro “nonce”).

Si el servidor que solicita la autenticación es un registrar, un *proxy redirect* o un UAS, la respuesta es la 401 (Unauthorized). Incluye obligatoriamente la cabecera WWW-Authenticate.

Si el servidor que solicita la autenticación es un proxy la respuesta es la 407 (*Proxy Authentication Required*). Incluye obligatoriamente la cabecera *Proxy-Authenticate*.

Como respuesta al “challenge”, el UAC envía de nuevo la petición, incrementando el valor de la cabecera Cseq, e incluyendo la acreditación en una cabecera: *Autorización* si la respuesta recibida fue la 401, o *Proxy-Autorización* si la respuesta recibida fue la 407. Estas cabeceras incluyen, en el parámetro “response” la identidad privada del usuario y password codificados.

El servidor analiza el contenido de la cabecera *Autorización* o *proxy-Autorización* verificando la validez de la acreditación. Si todo es correcto envía una respuesta 200 OK, que puede incluir la cabecera *Authentication-Info* (o *Proxy-Authentication-Info* si la autenticación la realiza un proxy), indicando información útil para nuevas autenticaciones, como el nonce que debe usarse (parámetro “nextnonce”).

Proceso de autenticación:

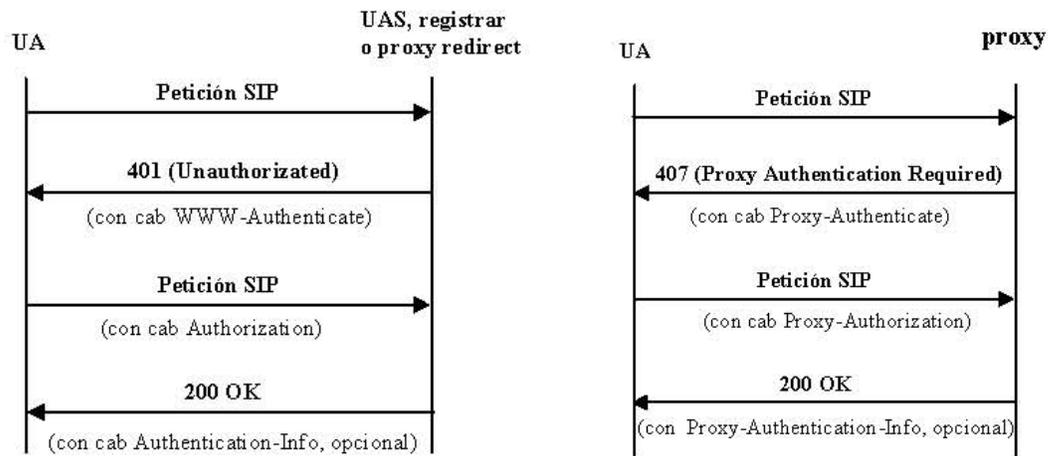


Figura 9: Proceso de Autenticación del Mensaje SIP

3.4.3 Dificultades Con Los Procesos de Traducción de Direcciones NAT Y PAT.

El supuesto agotamiento de los rangos de direcciones IPs utilizables en Internet ha obligado a utilizar direcciones IP privadas dentro de las redes de las empresas y usuarios domésticos.

Un equipo IP para ser alcanzado en Internet debe utilizar una IP pública para sus comunicaciones. Es necesario por tanto enmascarar la red interna en una o varias IPs públicas; para lo cual se utiliza NAT o PAT.

NAT es un mecanismo para conservar direcciones IP registradas en las grandes redes y simplificar las tareas de administración de direccionamiento IP. Mientras se enruta un paquete a través de un dispositivo de red, por lo general un firewall o router fronterizo, la dirección IP fuente se traduce de una dirección de red interna privada a una dirección IP pública enrutable. Esto permite que se transporte el paquete a través de redes externas públicas como la Internet. La dirección pública de la respuesta se traduce de nuevo a la dirección interna privada para su entrega dentro de la red interna. Una variación de NAT, conocida como Traducción de direcciones de puerto (PAT), permite la traducción de muchas direcciones privadas internas con una sola dirección pública externa.

3.4.3 .1 Protocolos en Tiempo Real y PAT

En Internet, las conversaciones que usan señalización de tipo SIP resultan en flujo constante de paquetes de pequeño tamaño entre los comunicantes. Estos paquetes de voz hacen uso de otro protocolo llamado RTP. El protocolo de transporte de tiempo real o Realtime Transport Protocol (RTP) es el encargado de llevar las conversaciones (la voz) de un lado a otro. En el RTP se define un mecanismo estándar para enviar audio y vídeo en Internet. De la misma forma que en una conversación existen dos flujos de voz, en una conversación en una red IP tenemos dos flujos de paquetes RTP.

Los Network Address Translators (NATs) son los grandes enemigos del RTP. Una red con un NAT consiste en varios ordenadores compartiendo, con el mundo exterior, un pool de dirección IP pública. Las máquinas situadas dentro de la red NAT usan direcciones privadas. Aunque el NAT permite conectar más fácilmente ordenadores a la red, lo hace al precio de no permitir una conexión puramente bidireccional. El efecto de un NAT con traducciones de Puerto (llamado PAT) en voz sobre IP es que no se pueden recibir conexiones iniciadas desde el exterior.

Existen varios problemas relacionados con PAT y VoIP. El más común de los problemas es conocido como “audio en una sola dirección” (oneway audio). Una conversación está compuesta por dos flujos de paquetes RTP distintos. En presencia de un PAT, sólo el flujo de a dentro hacia a fuera no es bloqueado; el flujo de a fuera hacia a dentro no tiene la misma suerte y puede atravesar el PAT. La consecuencia: el que inicia la llamada desde dentro del PAT no puede escuchar a la otra parte. Si los dos comunicantes se encuentran dentro de PATs las cosas se complican aún más, hasta el punto de que ningún flujo de audio llega a su destino final.

Este es uno de los problemas que presenta el protocolo SIP, que no tiene un adecuado direccionamiento de información para el funcionamiento de Traducción de direcciones de puerto PAT.

El protocolo SIP solo lleva la señalización de la llamada, mientras que el tráfico de audio (RTP) se intenta llevar a cabo de extremo a extremo. El problema viene por que el puerto al que se manda el audio es aleatorio. El router es capaz de dirigir correctamente la señalización, pero es incapaz de saber si el tráfico RTP es de esa llamada, y no sabe a donde mandarlo (debería mandárselo al mismo que la señalización). Como consecuencia, el audio no fluye correctamente entre los 2 interlocutores.

En un principio, puede ir todo bien, se inicia la llamada, se escucha el tono, y suena el teléfono de destino. El problema viene al descolgar, es decir, cuando el audio se establece entre los participantes de la conversación. En este punto, pueden pasar 2 cosas: que el llamante lo escuche a su interlocutor (el famoso “one way audio”) o que no haya audio en absoluto.

3.4.3.2 Esquema de la red SIP y el PAT

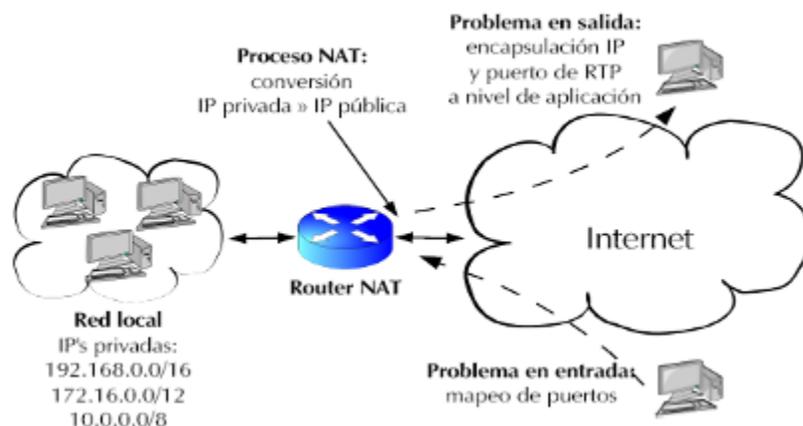


Figura 10: Dificultad que presenta el protocolo SIP al interactuar con una red LAN que utiliza NAT.

3.4.3.3 Posibles Soluciones a Los Problemas de SIP con PAT.

La solución a este tipo de problemas no suele ser fácil, ya que depende de varios factores: sin embargo podría solventarse con un servidor STUN, o también al enviar tanto señalización como streaming¹⁴ por el mismo flujo de datos (flujo UDP), mapeo de puertos, nat Helpers; se evitan los problemas derivados del PAT

Otra solución generada por cisco es **Cisco Secure PIX**.

➤ Servidor STUN

Un servidor STUN (Simple Traversal of User Datagram Protocol [UDP] a través de Network Address Translators (NATs), permite a los clientes NAT (tal como computadores detrás de un cortafuego), configurar llamadas telefónicas a un proveedor VoIP alojado afuera de su red local.

El servidor STUN permite a los clientes encontrar sus direcciones públicas, el tipo de PAT del cual están atrás y el puerto de Internet asociado por el PAT con el puerto

¹⁴ Streaming es un término que se refiere a ver u oír un archivo directamente en una página web sin necesidad de descargarlo antes al ordenador. Se podría describir como "hacer clic y obtener".

local específico. Esta información es usada para configurar comunicación UDP entre el cliente y el proveedor de VoIP para así establecer una llamada. El protocolo STUN está definido en el RFC 3489.

El servidor STUN es contactado en el puerto UDP 3478, sin embargo, el servidor indicará a los clientes que realicen pruebas en IP alternativas y también números de puertos (servidores STUN tienen 2 direcciones IP). El RFC indica que este puerto e IP son arbitrarios.

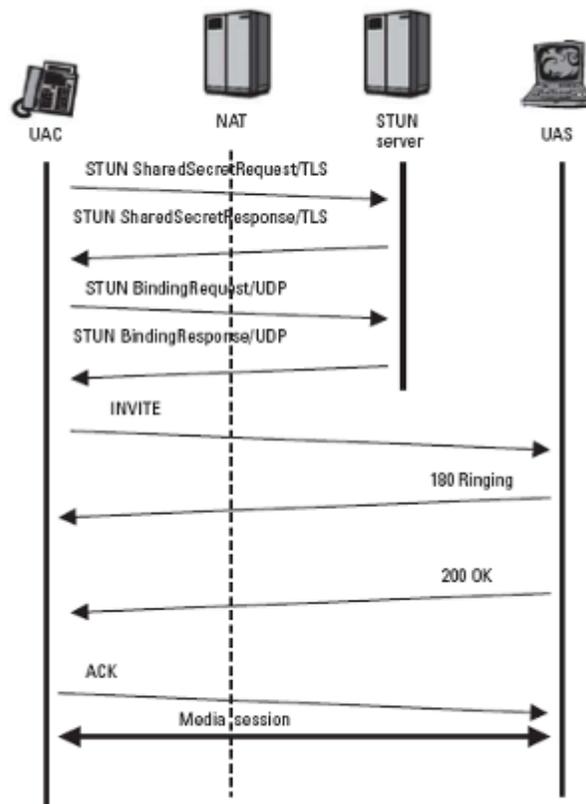


Figura 11: Esquema de funcionamiento del servidor STUN

➤ Cisco Secure PIX.

PIX es una de las soluciones de seguridad ofrecidas por Cisco Systems; se trata de un firewall completamente hardware: a diferencia de otros sistemas cortafuegos, PIX no se ejecuta en una máquina Unix, sino que incluye un sistema operativo empotrado

denominado Finesse que desde el espacio de usuario se asemeja más a un router que a un sistema Unix clásico.

PIX ha sido mejorado en su versión V6.2 para soportar PAT; La incorporada dirección IP y el puerto en el mensaje SIP/H.323 se traduce para el correcto PAT, y el correcto medio de conexiones son negocia durante la señalización es establecida.

PIX para PAT dinámico, modifica el valor de tiempo para que sea igual al valor de tiempo del registro del cliente, que es fijado cuando el cliente se registra con el servidor proxy SIP/ H.323 Gatekeeper.

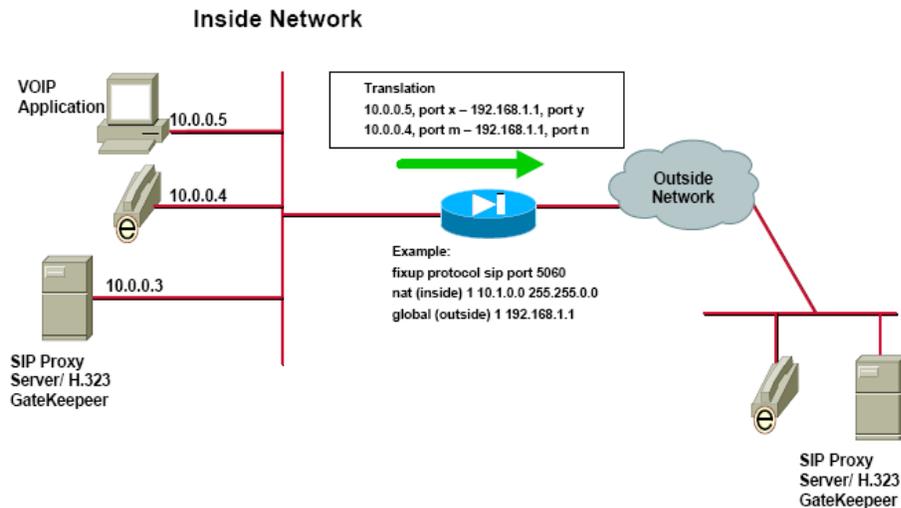


Figura 12: Esquema Del Funcionamiento de PIX Para PAT.

Para SIP, algunos mensajes han sido vistos con una dirección IP y no asociada al puerto, PIX no puede natear una dirección IP sin un puerto, por lo cual requiere PAT. PIX solo actualizara 5 PAT dinámicos por registro de cliente.

3.4.4 Operación de SIP En La Capa de Transporte

SIP es un protocolo perteneciente a la capa de aplicación del modelo OSI, pero su funcionamiento se basa en la comunicación de mensajes entre dispositivos de red. Para que este intercambio de comunicación se realice de forma eficiente es

importante que exista una interacción eficiente entre SIP y la capa de transporte. A continuación se define la forma de operar de los mensajes SIP.

3.4.4.1 Transacción Invite

Debido a que una transacción Invite desarrolla múltiples diálogos iniciales además de una solicitud ACK. Este mensaje tiene un tratamiento especial diferente a los otros métodos.

En muchos casos, para establecer una comunicación entre dos puntos de red, se debe pasar por más de un dispositivo, en el caso de una red SIP, se pasa por varios proxy y servidores. La conexión entre cada uno de estos servidores puede ser a través de protocolos que brinden seguridad de entrega, o también se utilizan protocolos que no brindan confiabilidad (UDP).

Para dotar de confiabilidad primero un AU envía el método INVITE al proxy; el proxy al recibir este método envía una respuesta temporal TRYING, mientras retransmite la solicitud al siguiente proxy. Estas respuestas temporales no poseen garantía de retransmisión. En cambio, al recibirse una respuesta de éxito (Mensaje 2XX), el mensaje es retransmitido por el mismo camino por el que se envió garantizando la entrega a través de retransmisiones.

Cuando la respuesta es un mensaje de no éxito, se realiza un proceso similar. Los dispositivos intermedios retransmiten el mensaje hasta recibir un ACK que confirme la recepción del mensaje.

3.4.4.2 Transacción Cancel

Cuando un AU envía un CANCEL al proxy, este envía una respuesta final, en este punto la sesión ya ha finalizado con este AU. En cambio el proxy envía un CANCEL al siguiente proxy o AU.

3.4.4.3 Otras Transacciones

Las tracciones restantes son tratadas de manera común. Los mensajes OPTION BYE y REGISTER son tratadas de manera similar a la tracción INVITE para obtener confiabilidad.

3.5 Aplicaciones de SIP

SIP es un protocolo desarrollado con la intención de ser el estándar para la iniciación, modificación y finalización de sesiones interactivas de usuario donde intervienen elementos multimedia como la voz. Entre unas de las funciones de este protocolo se encuentran.

- ✓ Establecimiento de llamadas VoIP.
- ✓ Establecimiento de llamadas multimedia.
- ✓ Eventos de notificación (Suscribir/Notificar).
- ✓ Mensajería instantánea.
- ✓ Transporte de señalización.
- ✓ Movilidad (3G).

Aplicación para llamar ELOSip

Corresponde a un Softphone programado en Java, que funciona en forma de Applet.

Esta aplicación implementa el protocolo SIP para comunicaciones de telefonía IP.

Se debe firmar digitalmente, debido a las restricciones de seguridad que presentan los Applets para establecer una conexión remota a un servidor.

Por ejemplo, SIP utiliza un método de encapsulado basado en texto (MIME) que permite transmitir datos y aplicaciones junto a la llamada de voz, haciendo fácil el

envío de tarjetas de visita, fotos, archivos MP3, información codificada, etc. , durante una llamada.

Actualmente se pueden definir como mínimo 4 perfiles en las aplicaciones mas utilizadas del protocolo SIP para servicios de voz:

- Perfil básico, donde ambos abonados son abonados SIP.
- Perfil A, donde A es un abonado SIP móvil y el abonado B es un abonado de la Red PSTN.
- Perfil B, donde A es un abonado SIP fijo y el abonado B es un abonado de la Red PSTN.
- Perfil C, donde tanto A como B son abonados de la Red PSTN y la Red IP es usada como tránsito entre dos redes PSTN.

Mediante los perfiles A, B y C es posible diseñar una red IP móvil o fija cuyos terminales de abonado utilicen el protocolo SIP y brinden los mismo servicios que brindaría una red PSTN así realizar o recibir llamadas hacia abonados de una red PSTN tradicional.

Todo lo anterior ha permitido crear las condiciones para:

- Crear redes formadas por pasarelas que utilicen el protocolo SIP y brinden los mismos servicios básicos y suplementarios de las redes PSTN así como la posibilidad de recibir o realizar llamadas hacia abonados de otras redes incluyendo PSTN.
- Utilizar la Red IP como tránsito para llamadas entre dos o mas redes PSTN.

3.6 Ejemplo de Comunicación SIP

Comunicación Entre Dos Personas Apartir De La Tecnologia VoIP

Escenario: dos personas comunicandose a través de un servidor SIP

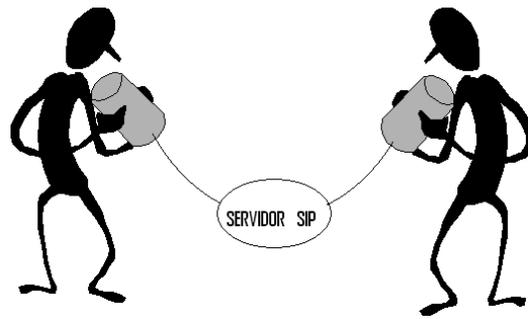
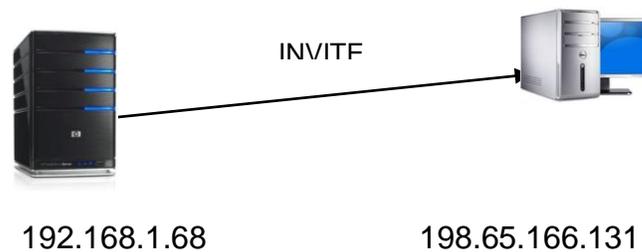


Figura 13: Dos personas comunicándose a partir del servidor SIP.

Persona Quien Realiza La Llamada

Establecimiento de Llamada.

Invite: El servidor (sandra.liss 192.168.1.254) envía una solicitud de conexión al agente de usuario (northamerica.sipphone 198.65.166.131).

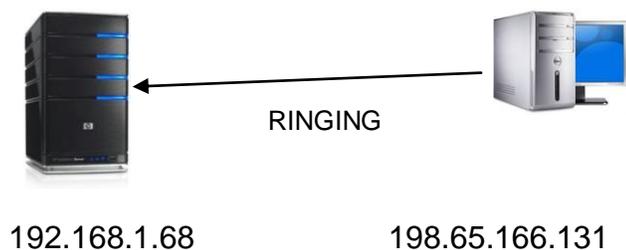


No.	Time	Source	Destination	Protocol	Info
216	18.038176	northamerica.sipphone	sandra.liss.local	SIP	status: 401 Unauthorized (0 bindings)
231	18.115485	sandra.liss.local	northamerica.sipphone	SIP	Request: REGISTER sip:proxy01.sipphone.com
261	18.414645	northamerica.sipphone	sandra.liss.local	SIP	status: 200 OK (1 bindings)
1746	359.748983	sandra.liss.local	northamerica.sipphone	SIP/SOF	Request: INVITE sip:ml.madrugos@proxy01.sipphone.com, with session desc...
1754	361.062241	northamerica.sipphone	sandra.liss.local	SIP	status: 100 trying -- your call is important to us
1766	361.752134	northamerica.sipphone	sandra.liss.local	SIP	status: 180 Ringing
1817	368.841269	northamerica.sipphone	sandra.liss.local	SIP/SOF	status: 200 OK, with session description
1832	369.247424	sandra.liss.local	northamerica.sipphone	SIP	Request: ACK sip:17472571557@190.87.116.223:64064;nat=yes
1858	369.811234	northamerica.sipphone	sandra.liss.local	SIP/SOF	status: 200 OK, with session description
1871	370.242645	sandra.liss.local	northamerica.sipphone	SIP	Request: ACK sip:17472571557@190.87.116.223:64064;nat=yes

Frame 1746 (1042 bytes on wire, 1042 bytes captured)					
Ethernet II, Src: sandra.liss.local (00:16:ec:76:d0:0a), Dst: 192.168.1.254 (00:14:9f:88:5e:e1)					
Internet Protocol, Src: sandra.liss.local (192.168.1.68), Dst: northamerica.sipphone.com (198.65.166.131)					
User Datagram Protocol, Src Port: 64064 (64064), Dst Port: 510 (5060)					
Session Initiation Protocol					

Figura 14: Establecimiento de la llamada a partir del servidor SIP

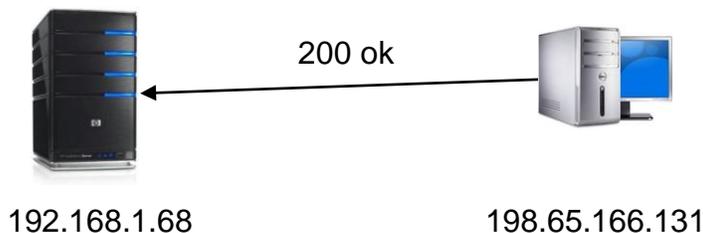
Ringin: El agente de usuario responde con un mensaje de respuesta temporal.



```
Frame 1766 (574 bytes on wire, 574 bytes captured)
Ethernet II, Src: 192.168.1.254 (00:14:95:88:5e:e1), Dst: sandraliss.local (00:16:ec:76:d0:0a)
Internet Protocol, Src: northamerica.sipphone.com (198.65.166.131), Dst: sandraliss.local (192.168.1.68)
User Datagram Protocol, Src Port: sip (5060), Dst Port: 64064 (64064)
Session Initiation Protocol
  Status-Line: SIP/2.0 180 Ringin
  Message Header
    Via: SIP/2.0/UDP 192.168.1.68:64064;received=190.87.53.157;branch=z9hG4bK-d87543-6b38a36ddb79372f-1--d87543-;rport=64064
    Record-Route: <sip:198.65.166.131;lr;ftag=d7531e5d>
    Contact: <sip:17472571557@190.87.116.223:64064>
    To: <sip:mel_rodrigo@proxy01.sipphone.com;tag=2b77263b>
    From: <sip:17472578630@proxy01.sipphone.com;tag=d7531e5d>
    Call-ID: b73bc06a5f4def32128576287208c2FuZHNjbG1zcw..
    CSeq: 1 INVITE
    User-Agent: WinGizmo/2.0.02 (Gizmo-s2n1)
    Content-Length: 0
    CQBM: 244
    RemoteIP: 190.87.116.223
```

Figura 15: Mensaje de Respuesta a la petición

Establecimiento: El agente usuario notifica al servidor que ha aceptado establecer la comunicacin con un 200 ok.



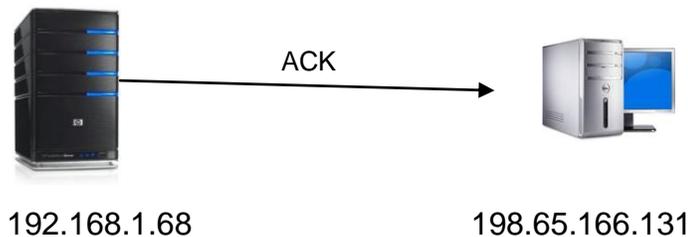
No. -	Time	Source	Destination	Protocol	Info
261	18.414645	northamerica.sipphone	sandraliss.local	SIP	Status: 200 OK (1 bindings)
1746	360.746983	sandraliss.local	northamerica.sipphone	SIP/SDP	Request: INVITE sip:mel_rodrigo@proxy01.sipphone.com, with session description
1754	361.062241	northamerica.sipphone	sandraliss.local	SIP	Status: 100 trying -- your call is important to us
1766	361.752134	northamerica.sipphone	sandraliss.local	SIP	Status: 180 Ringing
1817	368.841299	northamerica.sipphone	sandraliss.local	SIP/SDP	Status: 200 OK, with session description
1832	369.247424	sandraliss.local	northamerica.sipphone	SIP	Request: ACK sip:17472571557@190.87.116.223:64064;nat=yes
1858	369.811234	northamerica.sipphone	sandraliss.local	SIP/SDP	Status: 200 OK, with session description
1871	370.242645	sandraliss.local	northamerica.sipphone	SIP	Request: ACK sip:17472571557@190.87.116.223:64064;nat=yes
73012	1639.111105	sandraliss.local	northamerica.sipphone	SIP	Request: REGISTER sip:proxy01.sipphone.com
73020	1639.609833	sandraliss.local	northamerica.sipphone	SIP	Request: REGISTER sip:proxy01.sipphone.com

+ Frame 1817 (1108 bytes on wire, 1108 bytes captured)					
+ Ethernet II, Src: 192.168.1.254 [00:14:95:88:5e:e1], Dst: sandraliss.local (00:16:ec:76:d0:0a)					
+ Internet Protocol, Src: northamerica.sipphone.com (198.65.166.131), Dst: sandraliss.local (192.168.1.68)					
+ User Datagram Protocol, Src Port: sip (5060), Dst Port: 64064 (64064)					
+ Session Initiation Protocol					

+ Frame 1817 (1108 bytes on wire, 1108 bytes captured)					
+ Ethernet II, Src: 192.168.1.254 [00:14:95:88:5e:e1], Dst: sandraliss.local (00:16:ec:76:d0:0a)					
+ Internet Protocol, Src: northamerica.sipphone.com (198.65.166.131), Dst: sandraliss.local (192.168.1.68)					
+ User Datagram Protocol, Src Port: sip (5060), Dst Port: 64064 (64064)					
+ Session Initiation Protocol					
+ Status-Line: SIP/2.0 200 OK					
+ Message Header					
+ Via: SIP/2.0/UDP 192.168.1.68:64064;received=190.87.116.223;branch=z9hG4bK-d87643-6b7Aa36ddb79372f-1--d87643-;rport=64064					
+ Record-Route: < sip:198.65.166.131;lr;tag=d7531e5d >					
+ Contact: < sip:17472571557@190.87.116.223:64064;nat=yes >					
+ To: < sip:mel_rodrigo@proxy01.sipphone.com;tag=8b0b5932 >					
+ From: < sip:17472578630@proxy01.sipphone.com;tag=d7531e5d >					
+ Call-ID: b73bc06a5f4de132128576287200e2Fu2H1hbG1zvw..					
+ CSeq: 1 INVITE					
+ Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER					
+ Content-Type: application/sdp					
+ User-Agent: WinGizmo/2.0.02 (Gizmo-s2nl)					
+ Content-Length: 415					
+ JabberID: mel_rodrigo@chat.gizmo-project.com					
+ RemoteIP: 190.87.116.223					
+ Message Body					
+ Session Description Protocol					
+ Session Description Protocol Version (V): 0					
+ Owner/Creator, Session Id (O): GizmoProject 681009620 1 IN IP4 190.87.116.223					
+ Session Name (S): GizmoAudioSession					
+ Connection Information (C): IN IP4 190.87.116.223					
+ Time Description, active time (T): 0 0					
+ Media Description, name and address (M): audio/5004 RTP/AVP 103 102 0 8 3 106 96					
+ Media Attribute (A): rtpmap:103 15AC/16000					
+ Media Attribute (A): rtpmap:102 1L8C/8000					
+ Media Attribute (A): rtpmap:0 PCMU/8000					

Figura 16: El envío de un 200 ok, indica que se ha establecido la comunicación

Reconocimiento: el servidor responde con un mensaje ACK.



No. -	Time	Source	Destination	Protocol	Info
261	18.414645	northamerica.sipphone	sandraliss.local	SIP	Status: 200 OK (1 bindings)
1746	360.746983	sandraliss.local	northamerica.sipphone	SIP/SDP	Request: INVITE sip:me1_rodriigo@proxy01.sipphone.com, with session descr
1754	361.062241	northamerica.sipphone	sandraliss.local	SIP	Status: 100 trying -- your call is important to us
1766	361.752134	northamerica.sipphone	sandraliss.local	SIP	Status: 180 Ringing
1817	368.841299	northamerica.sipphone	sandraliss.local	SIP/SDP	Status: 200 OK, with session description
1832	369.247424	sandraliss.local	northamerica.sipphone	SIP	Request: ACK sip:17472571557@190.87.116.223:64064;nat=yes
1858	369.811234	northamerica.sipphone	sandraliss.local	SIP/SDP	Status: 200 OK, with session description
1871	370.242645	sandraliss.local	northamerica.sipphone	SIP	Request: ACK sip:17472571557@190.87.116.223:64064;nat=yes
3012	1639.111105	sandraliss.local	northamerica.sipphone	SIP	Request: REGISTER sip:proxy01.sipphone.com
3020	1639.609833	sandraliss.local	northamerica.sipphone	SIP	Request: REGISTER sip:proxy01.sipphone.com

+ Frame 1832 (552 bytes on wire, 552 bytes captured)					
+ Ethernet II, Src: sandraliss.local (00:16:ec:76:d0:0a), Dst: 192.168.1.254 (00:14:95:88:5e:e1)					
+ Internet Protocol, Src: sandraliss.local (192.168.1.68), Dst: northamerica.sipphone.com (198.65.166.131)					
+ User Datagram Protocol, Src Port: 64064 (64064), Dst Port: sip (5060)					
+ Session Initiation Protocol					

+ Frame 1832 (552 bytes on wire, 552 bytes captured)					
+ Ethernet II, Src: sandraliss.local (00:16:ec:76:d0:0a), Dst: 192.168.1.254 (00:14:95:88:5e:e1)					
+ Internet Protocol, Src: sandraliss.local (192.168.1.68), Dst: northamerica.sipphone.com (198.65.166.131)					
+ User Datagram Protocol, Src Port: 64064 (64064), Dst Port: sip (5060)					
+ Session Initiation Protocol					
- Request-Line: ACK sip:17472571557@190.87.116.223:64064;nat=yes SIP/2.0					
Method: ACK					
[Resent Packet: False]					
- Message Header					
+ Via: SIP/2.0/UDP 192.168.1.68:64064;branch=z9hG4bK-d87543-382b6e0e97407765-1--d87543-;rport					
Max-Forwards: 70					
Route: <sip:198.65.166.131;lr;ftag=d7531e5d>					
+ Contact: <sip:17472578630@190.87.53.157:64064>					
+ To: <sip:me1_rodriigo@proxy01.sipphone.com>;tag=5b0b5932					
+ From: <sip:17472578630@proxy01.sipphone.com>;tag=d7531e5d					
Call-ID: b73bc06a5f4def3212857628720@c2FuZHJhbG1zcw..					
+ CSeq: 1 ACK					
User-Agent: WinGizmo/2.0.02 (Gizmo-s2n1)					
Content-Length: 0					

Figura 17: Cuando se ha establecido la comunicación el servidor responde con un ACK

Finalización de Llamada.

BYE: El proxy transmite la solicitud de terminación de llamada.

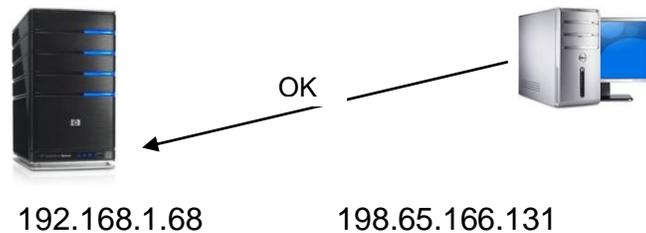


No.	Time	Source	Destination	Protocol	Info
71060	1641.323606	northamerica.sipphone	sandnalliss.local	SIP	Status: 200 OK (1 bindings)
162110	3261.476078	sandnalliss.local	northamerica.sipphone	SIP	Request: REGISTER sip:proxy01.sipphone.com
162116	3261.705416	northamerica.sipphone	sandnalliss.local	SIP	Status: 401 Unauthorized (0 bindings)
162148	3262.005561	sandnalliss.local	northamerica.sipphone	SIP	Request: REGISTER sip:proxy01.sipphone.com
162164	3262.215529	northamerica.sipphone	sandnalliss.local	SIP	Status: 200 OK (1 bindings)
212197	4064.829484	sandnalliss.local	northamerica.sipphone	SIP	Request: BYE sip:174723715578190.87.116.233:64064;nat=yes
212219	4065.328101	sandnalliss.local	northamerica.sipphone	SIP	Request: BYE sip:174723715578190.87.116.233:64064;nat=yes
212248	4065.860146	northamerica.sipphone	sandnalliss.local	SIP	Status: 200 OK
212411	4128.634487	sandnalliss.local	northamerica.sipphone	SIP	Request: REGISTER sip:proxy01.sipphone.com
212442	4130.055416	northamerica.sipphone	sandnalliss.local	SIP	Status: 401 Unauthorized (0 bindings)

Frame 212197 (552 bytes on wire, 552 bytes captured)
 Ethernet II, Src: sandnalliss.local (00:16:ec:76:d0:0a), Dst: 192.168.1.254 (00:14:9f:88:5e:e1)
 Internet Protocol, Src: sandnalliss.local (192.168.1.68), Dst: northamerica.sipphone.com (198.65.166.131)
 User Datagram Protocol, Src Port: 64064 (64064), Dst Port: sip (5060)
 Session Initiation Protocol

Figura 18: Petición BYE que es enviada cuando se quiere finalizar la llamada.

OK: El agente usuario envía un mensaje para aceptar la finalización de la sesión.



No.	Time	Source	Destination	Protocol	Info
71060	1641.323606	northamerica.sipphone	sandnalliss.local	SIP	Status: 200 OK (1 bindings)
162110	3261.476078	sandnalliss.local	northamerica.sipphone	SIP	Request: REGISTER sip:proxy01.sipphone.com
162116	3261.705416	northamerica.sipphone	sandnalliss.local	SIP	Status: 401 Unauthorized (0 bindings)
162148	3262.005561	sandnalliss.local	northamerica.sipphone	SIP	Request: REGISTER sip:proxy01.sipphone.com
162164	3262.215529	northamerica.sipphone	sandnalliss.local	SIP	Status: 200 OK (1 bindings)
212197	4064.829484	sandnalliss.local	northamerica.sipphone	SIP	Request: BYE sip:174723715578190.87.116.233:64064;nat=yes
212219	4065.328101	sandnalliss.local	northamerica.sipphone	SIP	Request: BYE sip:174723715578190.87.116.233:64064;nat=yes
212248	4065.860146	northamerica.sipphone	sandnalliss.local	SIP	Status: 200 OK
212411	4128.634487	sandnalliss.local	northamerica.sipphone	SIP	Request: REGISTER sip:proxy01.sipphone.com
212442	4130.055416	northamerica.sipphone	sandnalliss.local	SIP	Status: 401 Unauthorized (0 bindings)

Frame 212248 (563 bytes on wire, 563 bytes captured)
 Ethernet II, Src: 192.168.1.254 (00:14:9f:88:5e:e1), Dst: sandnalliss.local (00:16:ec:76:d0:0a)
 Internet Protocol, Src: northamerica.sipphone.com (198.65.166.131), Dst: sandnalliss.local (192.168.1.68)
 User Datagram Protocol, Src Port: sip (5060), Dst Port: 64064 (64064)
 Session Initiation Protocol

```

Frame 212248 (563 bytes on wire, 563 bytes captured)
Ethernet II, Src: 192.168.1.254 (00:14:95:88:5e:e1), Dst: sandraliss.local (00:16:ec:76:d0:0a)
Internet Protocol, Src: northamerica.sipphone.com (198.65.166.131), Dst: sandraliss.local (192.168.1.68)
User Datagram Protocol, Src Port: sip (5060), Dst Port: 64064 (64064)
Session Initiation Protocol
Status-Line: SIP/2.0 200 OK
  Status-Code: 200
  [Resent Packet: False]
Message Header
  Via: SIP/2.0/UDP 192.168.1.68:64064;received=190.87.53.157;branch=z9hG4bK-d87543-b26b0060140e2803-1--d87543-;rport=64064
  Record-Route: <sip:198.65.166.131;r;ftag=d7531e5d>
  Contact: <sip:17472571557@190.87.116.223:64064;nat=yes>
  To: <sip:mel_rodrigo@proxy01.sipphone.com;tag=5b0b5932>
  From: <sip:17472578630@proxy01.sipphone.com;tag=d7531e5d>
  Call-ID: b73bc06a5f4def32128576287208c2FuZHjhbG1zcw..
  CSeq: 2 BYE
  User-Agent: WinGizmo/2.0.02 (Gizmo-s2n1)
  Content-Length: 0
  RemoteIP: 190.87.116.223

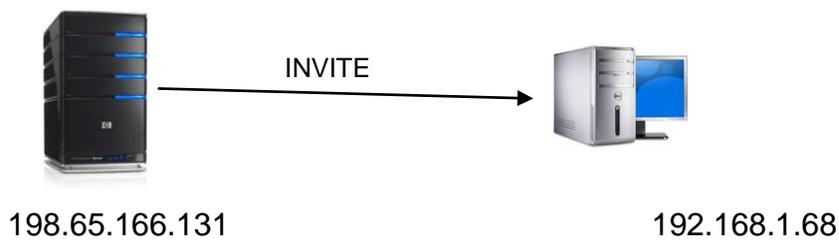
```

Figura 19: Se acepta finalización de la llamada con un OK

Persona Quien Acepta La Llamada

Establecimiento de llamada.

Invite: El servidor envia una solicitud de conexion al agente de usuario.



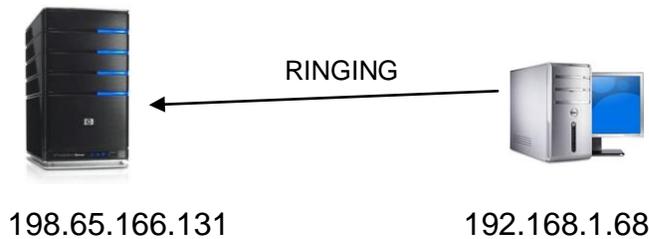
```

Session Initiation Protocol
Request-Line: INVITE sip:17472571557@190.87.116.223:64064 SIP/2.0
Method: INVITE
[Resent Packet: False]
Message Header
Record-Route: <sip:198.65.166.131;f-tag=d7531e5d;lr>
Via: SIP/2.0/UDP 198.65.166.131;branch=z9hG4bK0d88.b8f81592.1
Via: SIP/2.0/UDP 192.168.1.68:64064;received=190.87.53.157;branch=z9hG4bK-d87543-6b38a36ddb79372f-1--d87543-;rport=64064
Max-Forwards: 16
Contact: <sip:17472578630@190.87.53.157:64064>
To: <sip:me1_rodrigo@proxy01.sipphone.com>
From: <sip:17472578630@proxy01.sipphone.com;tag=d7531e5d>
Call-ID: b73bc06a5f4def3212857628720@c2FuZHIhOG12ow..
CSeq: 1 INVITE
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER
Content-Type: application/sdp
User-Agent: WinGizmo/2.0.02 (Gizmo-s2n1)
Content-Length: 414
JabberID: sandraliss@chat.gizmo-project.com
CQBM: 244
RemoteIP: 190.87.53.157
P-contact: 3

```

Figura 20: Servidor envía Un INVITE al usuario solicitando la conexión

Ringin: El agente de usuario responde con un mensaje de respuesta temporal.



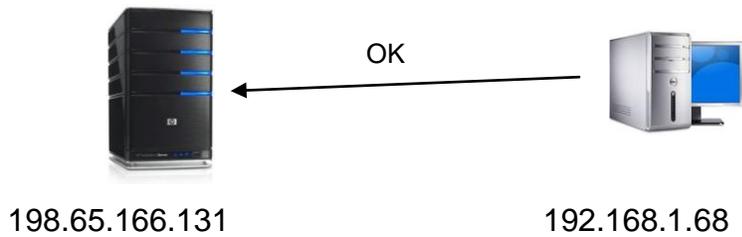
```

Session Initiation Protocol
Status-Line: SIP/2.0 180 Ringin
Status-Code: 180
[Resent Packet: False]
Message Header
Via: SIP/2.0/UDP 198.65.166.131;branch=z9hG4bK0d88.b8f81592.1
Via: SIP/2.0/UDP 192.168.1.68:64064;received=190.87.53.157;branch=z9hG4bK-d87543-6b38a36ddb79372f-1--d87543-;rport=64064
Record-Route: <sip:198.65.166.131;lr;f-tag=d7531e5d>
Contact: <sip:17472571557@proxy01.sipphone.com>
To: <sip:me1_rodrigo@proxy01.sipphone.com;tag=2b77263b>
From: <sip:17472578630@proxy01.sipphone.com;tag=d7531e5d>
Call-ID: b73bc06a5f4def3212857628720@c2FuZHIhOG12ow..
CSeq: 1 INVITE
User-Agent: WinGizmo/2.0.02 (Gizmo-s2n1)
Content-Length: 0
CQBM: 244

```

Figura 21: El usuario envía una respuesta temporal

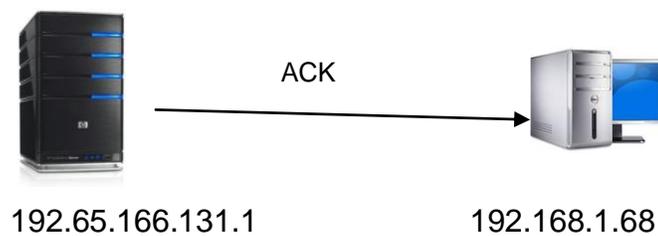
Establecimiento: El agente usuario notifica al servidor que ha aceptado establecer la comunicación.



```
Session Initiation Protocol
Status-Line: SIP/2.0 200 OK
Status-Code: 200
[Resent Packet: False]
Message Header
Via: SIP/2.0/UDP 198.65.166.131;branch=z9hG4bK0d88.b8f81592.1
Via: SIP/2.0/UDP 192.168.1.68:64064;received=190.87.53.157;branch=z9hG4bK-d87543-6b38a36ddb79372f-1--d87543-;rport=64064
Record-Route: <sip:198.65.166.131;r;ftag=d7531e5d>
Contact: <sip:17472571557@190.87.116.223:64064>
To: <sip:me1_rodrigo@proxy01.sipphone.com;tag=5b0b5932>
From: <sip:17472578630@proxy01.sipphone.com;tag=d7531e5d>
Call-ID: b73bc06a5f4def3212857628720@c2FUZHjhbG1zcw..
CSeq: 1 INVITE
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER
Content-Type: application/sdp
User-Agent: WinGizmo/2.0.02 (Gizmo-s2n1)
Content-Length: 415
JabberID: me1_rodrigo@chat.gizmoproject.com
Message Body
```

Figura 22: Respuesta OK, notifica que se ha establecido la comunicación.

Reconocimiento: el servidor responde con un mensaje ACK.

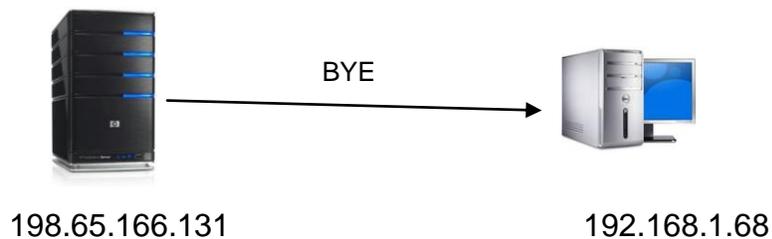


```
Session Initiation Protocol
Request-Line: ACK sip:17472571557@190.87.116.223:64064 SIP/2.0
Method: ACK
[Resent Packet: False]
Message Header
Record-Route: <sip:198.65.166.131;ftag=d7531e5d;r>
Via: SIP/2.0/UDP 198.65.166.131;branch=0
Via: SIP/2.0/UDP 192.168.1.68:64064;received=190.87.53.157;branch=z9hG4bK-d87543-382b6e0e97407765-1--d87543-;rport=64064
Max-Forwards: 16
Contact: <sip:17472578630@190.87.53.157:64064>
To: <sip:mel_rodrigo@proxy01.sipphone.com;tag=5b0b5932>
From: <sip:17472578630@proxy01.sipphone.com;tag=d7531e5d>
Call-ID: b73bc06a5f4def3212857628720dc2FuZHjhbG12cw..
CSeq: 1 ACK
User-Agent: WinGizmo/2.0.02 (Gizmo-s2n1)
Content-Length: 0
P-hint: nr-enforced
P-NATED-URI: YES (1)
```

Figura 23: El servidor responde con un mensaje de la petición ACK

Finalización de llamada.

BYE: El proxy transmite la solicitud de terminación de llamada.



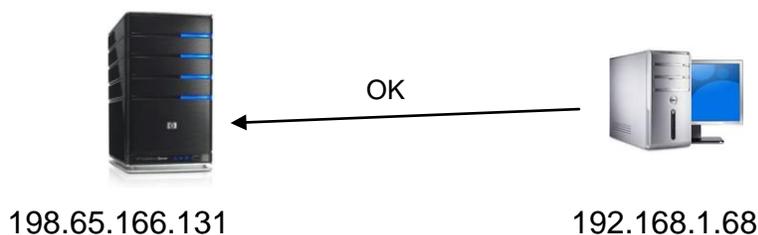
```

Session Initiation Protocol
Request-Line: BYE sip:17472571557@190.87.116.223:64064 SIP/2.0
Method: BYE
[Resent Packet: False]
Message Header
Record-Route: <sip:198.65.166.131;ftag=d7531e5d;r>
Via: SIP/2.0/UDP 198.65.166.131;branch=z9hG4bKdc88.12b5d304.0
Via: SIP/2.0/UDP 192.168.1.68:64064;received=190.87.53.157;branch=z9hG4bK-d87543-b26b0060140e2803-1--d87543-;rport=64064
Max-Forwards: 16
Contact: <sip:17472578630@190.87.53.157:64064>
To: <sip:me1_rodrigo@proxy01.sipphone.com;tag=5b0b5932>
From: <sip:17472578630@proxy01.sipphone.com;tag=d7531e5d>
Call-ID: b73bc06a5f4def3212857628720@c2FuZlJhbG1zcw..
CSeq: 2 BYE
User-Agent: WinGizmo/2.0.02 (Gizmo-s2n1)
Content-Length: 0
RemoteIP: 190.87.53.157
P-hint: rr-enforced
P-NATed-URI: YES (1)

```

Figura 24: Cuando finaliza la comunicación el servidor transmite la solicitud a través de la petición BYE

OK: el agente usuario envía un mensaje para aceptar la finalización de la sesión.



```

Session Initiation Protocol
Status-Line: SIP/2.0 200 OK
Status-Code: 200
[Resent Packet: False]
Message Header
Via: SIP/2.0/UDP 198.65.166.131;branch=z9hG4bKdc88.12b5d304.0
Via: SIP/2.0/UDP 192.168.1.68:64064;received=190.87.53.157;branch=z9hG4bK-d87543-b26b0060140e2803-1--d87543-;rport=64064
Record-Route: <sip:198.65.166.131;r;ftag=d7531e5d>
Contact: <sip:17472571557@190.87.116.223:64064>
To: <sip:me1_rodrigo@proxy01.sipphone.com;tag=5b0b5932>
From: <sip:17472578630@proxy01.sipphone.com;tag=d7531e5d>
Call-ID: b73bc06a5f4def3212857628720@c2FuZlJhbG1zcw..
CSeq: 2 BYE
User-Agent: WinGizmo/2.0.02 (Gizmo-s2n1)
Content-Length: 0

```

Figura 25: El mensaje ok confirma que la comunicación ha finalizado

CAPITULO IV

SOLUCIÓN DE VoIP UTILIZANDO ASTERISK

Introducción

Las centrales telefónicas PBX, permiten a las empresas crear y administrar sus propias conexiones de telefonía internas y a la vez que estas puedan comunicarse con el exterior.

Para desplegar servicios de VoIP, existen múltiples proveedores como Cisco, Siemens, NorTel Etc, sin embargo todas estas obligan a utilizar sus propios protocolos, y que los diferentes dispositivos a utilizar en la red pertenezcan a un mismo proveedor. Esto sin mencionar los altos costos que implican. Es así como nace la central telefónica Asterisk y brinda todas las ventajas y funcionalidades de una PBX sin limitar el uso restringido de protocolos y equipo de un único proveedor.

Este capítulo se enfoca en dar conocer al lector la estructura y funcionamiento de Asterisk; además del conocimiento básico para comprender como este junto al protocolo de inicio de sesión SIP permiten desarrollar una completa solución de VoIP sin la necesidad de contar con PBX tradicional,

4.0 SOLUCIÓN DE VoIP UTILIZANDO ASTERISK

4.1 Definición y Generalidades

4.1.1 Private Branch Exchange - PBX.

PBX son las siglas de Private Branch Exchange. Una PBX se encarga de establecer conexiones entre terminales de una misma empresa, o de hacer que se cursen llamadas al exterior. Hace que las extensiones tengan acceso desde el exterior y desde el interior, y ellas a su vez tengan acceso también a otras extensiones y a una línea externa.

Una PBX se usa realmente para que los miembros de la organización a la que corresponda la PBX se puedan comunicar fácilmente entre ellos, incluso si se encuentran lejos de su puesto de trabajo habitual. Para ello se establece un número de teléfono que acepta las llamadas entrantes, y tras pedir un código (o sin pedir código), nos pide amablemente que pulsemos la extensión a la cual queremos llamar. Desde dentro de la PBX, también se pueden realizar llamadas al exterior

4.1.1.1 Esquema de la PBX

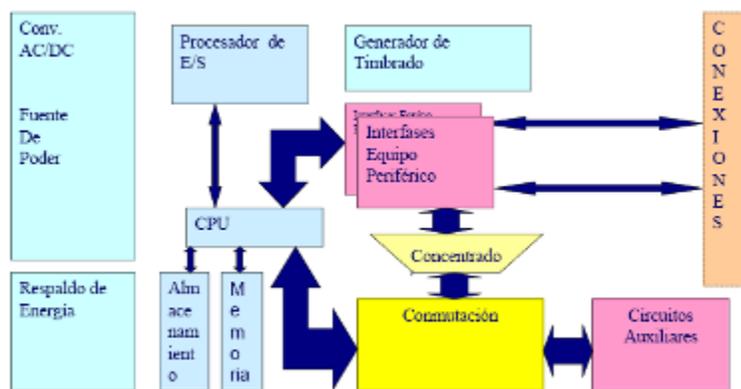


Figura 26: esquema de arquitectura de una PBX

4.1.1.2 Beneficio de las PBX

- Transferencia de Llamadas
- Conferencia
- Captura de llamadas
- Llamada en espera
- Servicio diurno y nocturno
- Clases de Servicio
- Acceso a red de parlantes
- Interfaces con teléfonos de puerta
- Restricciones de telediscado
- Desvíos de llamadas

4.1.1.3 Estructura de las PBX

Toda PBX consta de unos elementos hardware (procesadores, extensiones, etc) y otro software, siendo estos últimos los que definen la gama de servicios ofrecidos y que marcan las principales diferencias entre unas y otras.

Los principales componentes de la arquitectura de un sistema de conmutación PBX digital.

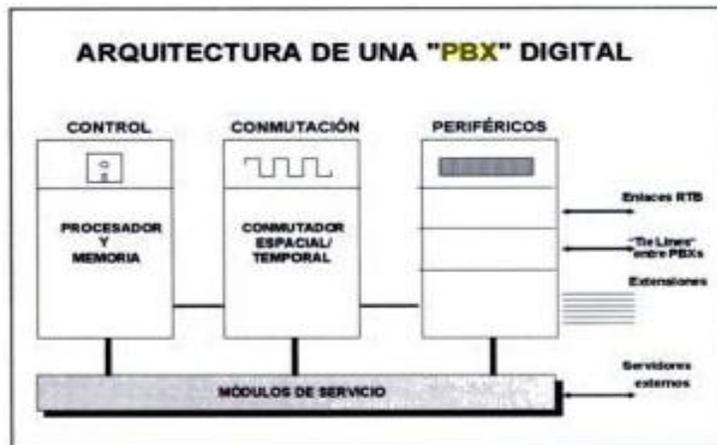


Figura27: Diagrama de bloques que muestra la unidad de conmutación y la de control.

El empleo de técnicas SPC ha hecho que el software sea tan importante o más que el hardware por lo que una división del sistema puede ser:

- a) Hardware para conectar a los usuarios y conmutar las llamadas.
- b) Software para controlar el hardware y dar a los usuarios las facilidades y los servicios deseados

4.1.2 Asterisk Como PBX

Asterisk es una aplicación software libre de una central telefónica (PBX). Como cualquier PBX, se puede conectar un número determinado de teléfonos para hacer llamadas entre sí e incluso conectar a un proveedor de VoIP o bien a una RDSI tanto básicos como primarios. Su nombre viene del símbolo asterisco (*) en inglés.

4.1.2.1 Esquema Conceptual

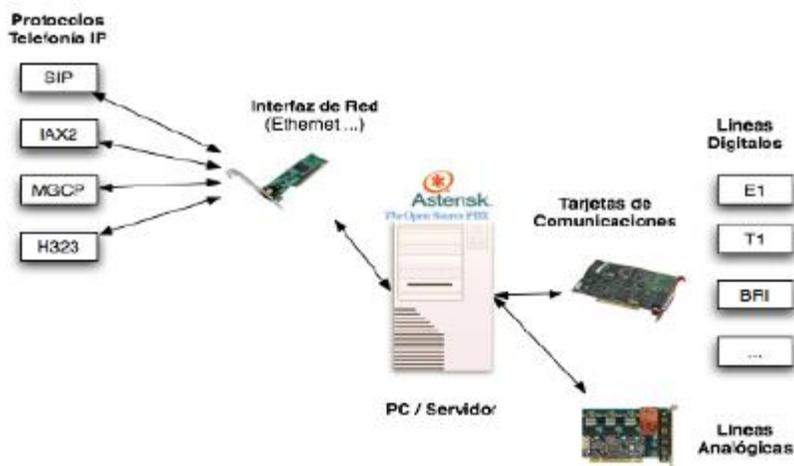


Figura 28: Esquema conceptual del funcionamiento de Asterisk

4.1.2.2 Aspectos Generales

- Asterisk es demonio un “demonio” que se ejecuta en segundo plano. Al igual que el resto de servidores conocidos (apache, openssl, proftpd).
- Asterisk hace voz sobre IP en cuatro protocolos y puede interoperar con casi todos los basados en estándares de telefonía utilizando el equipo de hardware relativamente barato.
- La configuración de Asterisk normalmente se almacena en varios ficheros de texto editables de forma tradicional que permiten al usuario poder interactuar con ellos según el ambiente donde se desarrolla.

4.2 Historia de Asterisk

La central telefónica Asterisk fue desarrollada por Mark Spencer un estudiante de ingeniería informática de la Universidad de Auburn, Alabama. Inicio con la creación de la empresa "Linux Support Services", que tenía como principal objetivo brindar soporte a usuarios de sistemas Linux, al verse en la necesidad de contar con una central telefónica y no contar con los recursos económicos suficientes decide crear su propia central con una PC y el lenguaje de programación C.

Por otro lado Jim Dixon había decidido que podía crear tarjetas telefónicas y drivers adecuados para evitar hacer uno de los costosos DSP¹⁵ (*Digital Signal Processor*) utilizados en las tarjetas de telefonía. Él considero que las nuevas CPU basadas en el procesador x86 de Intel, contaban con la potencia suficiente para realizar las complejas operaciones requeridas. De esta manera realiza sus primeros trabajos con una tarjeta Mitel MB89000C "ISDN Express Development" que 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 FreeBSD¹⁶.

Tras el éxito del primer driver, crea un nuevo diseño más eficiente de tarjeta ISA, que podía manejar 2 interfaces T1 utilizando una CPU Pentium III. Tiempo después, decide debido a las solicitudes de personas interesadas, desarrollar una versión del driver para Linux, utiliza RedHat 6.0 y para ello publica en Internet tanto el driver como el diseño de la tarjeta. Cuarenta y ocho horas más tarde, Mark Spencer contacta con él y se ofrece a reescribir el driver de Linux a fin de perfeccionarlo, logran obtener las herramientas necesarias para desarrollar una PBX por software completamente funcional, de esta manera nace el proyecto **Asterisk**.

¹⁵ **DSP**: es un sistema basado en un procesador o microprocesador que posee un juego de instrucciones, un hardware y un software optimizados para aplicaciones que requieran operaciones numéricas a muy alta velocidad.

¹⁶ **FreeBSD** es un sistema operativo libre para computadoras basado en las CPU de arquitectura Intel, incluyendo procesadores 386, 486 (versiones SX y DX), y Pentium.

En 1999 se crea la empresa Digium y se lanza la primera release Asterisk 0.1 que era licencia GPL2, posteriormente la versión 0.1.1 además de múltiples mejoras pasa a ser GPL¹⁷.

4.3 Características de Asterisk

4.3.1 Características Generales

- Asterisk incluye muchas características anteriormente sólo disponibles en caros sistemas propietarios PBX como: creación de extensiones, buzón de voz, conferencias, IVR¹⁸, distribución automática de llamadas, Registros de llamada detallados, y otras muchas más.
- Los usuarios pueden crear nuevas funcionalidades mediante el propio lenguaje de Asterisk o módulos escritos en C o mediante scripts AGI escritos en Perl o en otros lenguajes programación soportados por Linux.
- Soporta muchos protocolos VoIP como pueden ser SIP, H.323, IAX y MGCP. Asterisk puede interoperar con terminales IP actuando como un registrador y como Gateway entre ambos.
- Al soportar una mezcla de la telefonía tradicional y los servicios de VoIP, Asterisk permite a los desarrolladores construir nuevos sistemas telefónicos de forma eficiente o migrar de forma gradual los sistemas existentes a las nuevas tecnologías. Algunos sitios usan Asterisk para reemplazar a antiguas centralitas propietarias, otros para proveer funcionalidades adicionales y algunas otras para reducir costes en llamadas a larga distancia utilizando Internet.

¹⁷ **GPL**, es una licencia creada por la Free Software Foundation a mediados de los 80, y está orientada principalmente a proteger la libre distribución, modificación y uso de software.

¹⁸ **IVR**(Interactive Voice Response)basado en Asterisk permite la interacción con bases de datos internas del sistema (MySQL o PostgreSQL), externas como Oracle, DB2, Sybase, Informix, MS SQL Server o Web Services bajo SOAP.

- Compatibilidad con Telefonía Estándar:
 - E&M
 - E&M Wink
 - Feature Group D
 - FXS
 - FXO
 - GR-303
 - Loopstart
 - MF and DTMF support

4.3.2 Licenciamiento de Asterisk

Si bien originalmente fue desarrollado para el sistema operativo Linux, en la actualidad también funciona en BSD, MacOSX, Solaris y Microsoft Windows, aunque en su plataforma nativa Linux es la mejor soportada de todos.

Sin embargo existen principalmente dos versiones estas son:

- *La versión Open Source.*
- *Licencia Comercial*

La version Open Source:

Asterisk Business Edition: edición comercial.

- Se basa en la Open Source, pero se eliminan todos aquellos elementos susceptibles de causar problemas.
- Fuertemente probada.
- Al comprarla incluye soporte durante un año

Es software open source, desarrollado principalmente por la empresa americana DIGIUM.

- Liberado con la licencia GPL.
- Se ejecuta en PC estándar (arquitectura x86,x86_64, ppc) bajo GNU/Linux, BSD o MacOSX.

GPL (General Public License)

- Libertad de ejecutar el programa, con cualquier propósito.
- Libertad de modificar el programa (acceso al código fuente).

- Libertad de redistribuir copias.
- Libertad de mejorar el programa y publicar mejoras (junto al código fuente).
- Añadidos de terceros al código de Asterisk implican la cesión de los derechos comerciales a Digium pero se conserva el copyright del desarrollador. El Nuevo conjunto se libera como GPL.
-

Licencia Comercial

- Tiene un coste económico en función de las características requeridas (número de canales simultáneos, aplicaciones de software, etc.).
- El comprador recibe el código fuente y soporte de Digium.
- El comprador puede incorporar sus propios cambios en su Asterisk sin necesidad de publicarlos como GPL, y puede vender su producto a terceros.

Asterisk Business Edition

- Tiene un coste económico.
- Solución preparada para empresas:
 - CD de instalación con binarios y drivers.
 - Manuales.
 - Clave de autenticación para activación y soporte.
 - Soporte gratuito durante un año.
 - Hasta 40 llamadas simultáneas.
- Basado en Asterisk 1.2 con algunas características eliminadas.
- Todas las mejoras en la versión comercial son incluidas en la versión GPL.

4.3.3 Funcionalidades de Llamada

Asterisk dispone de un conjunto de funcionalidades ofrecidas tan solo por grandes sistemas PBX propietarios

- Receptor de Alarmas
- Autenticación (Autenticación)
- Respuesta automatizada
- Listas negras
- Registros de llamada detallados
- Desvío de llamada si la extensión está ocupado
- Desvío de llamada si la extensión no responde
- Monitorización de llamadas
- Llamadas en espera.
- Grabación de llamadas
- Recuperación de llamadas
- Escucha de llamadas
- Transferencia de llamadas
- Identificación del llamante
- Bloqueo de llamante
- Identificación del llamante durante la llamada en espera
- Conferencia de voz
- Marcación por nombre
- Llamadas de emergencia
- Recepción y transmisión de Fax
- Presentación interactiva del directorio
- Respuesta vocal interactiva (IVR)
- Agentes locales y remotos
- Música en espera y en transferencia
- Protocolo de establecimiento abierto
- Conversión de protocolos
- Captura de llamada remota
- Soporte de oficina remota
- Mensajería SMS
- Detección de habla
- Texto a Voz (TTS)
- Llamada a tres
- Transcodificación
- Trunking
- Pasarelas VoIP
- Buzón de Voz
- Indicador visual de mensaje en espera

4.3.4 Ventajas

- **Funcionalidad**

Asterisk dispone de todas las funcionalidades de las grandes centralitas propietarias (Cisco, Avaya, Alcatel, Siemens, etc). Desde las más básicas (desvíos, capturas, transferencias, multi-conferencias) hasta las más avanzadas (Buzones de voz, IVR, CTI, ACD).

- **Escalabilidad**

El sistema puede dar servicio desde 10 usuarios en una sede de una pequeña empresa, hasta 10.000 de una multinacional repartidos en múltiples sedes.

- **Competitividad en coste**

No solo por ser un sistema de código abierto (Open Source) sino gracias a su arquitectura hardware: utiliza plataforma servidor estándar (de propósito no específico) y tarjetas PCI para los interfaces de telefonía, que por la competencia del mercado se han ido abaratando progresivamente.

- **Interoperabilidad y Flexibilidad.**

Asterisk ha incorporado la mayoría de estándares de telefonía del mercado, tanto los tradicionales (TDM) con el soporte de puertos de interfaz analógicos (FXS y FXO) y RDSI (básicos y primarios), como los de telefonía IP (SIP, H.323, MGCP, SCCP/Skinny). Eso le permite conectarse a las redes públicas de telefonía tradicional e integrarse fácilmente con centralitas tradicionales (no IP) y otras centralitas IP.

- **Sistema de Telefonía Entre Sedes Remotas**

Una ventaja inmediata de las soluciones de telefonía IP consiste en la posibilidad de realizar llamadas entre sedes remotas sin coste e integrando el sistema de numeración (extensiones) entre ellas.

Asterisk ofrece una solución ideal para este tipo de organizaciones con sedes distantes.

Sistema Centralizado

Existe un servidor centralizado por donde entran y salen las llamadas a la red de telefonía tradicional, y que se encarga de gestionar los clientes de VoIP (extensiones VoIP) de todas las delegaciones.

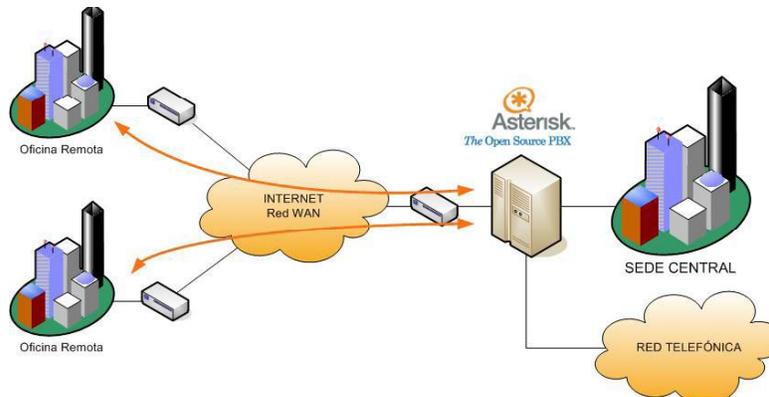


Figura 29: Ejemplo de sistema centralizado

Sistema distribuido

Cada delegación tiene su propia centralita Asterisk conectada a la red WAN de la empresa (A través de internet o líneas dedicadas) y a la red telefónica.

Entre los Asterisk, a través del protocolo IAX (Inter.- Asterisk eXchange) intercambian información de rutado, permitiendo una gestión del sistema de telefonía de la compañía y un rutado inteligente de llamadas.

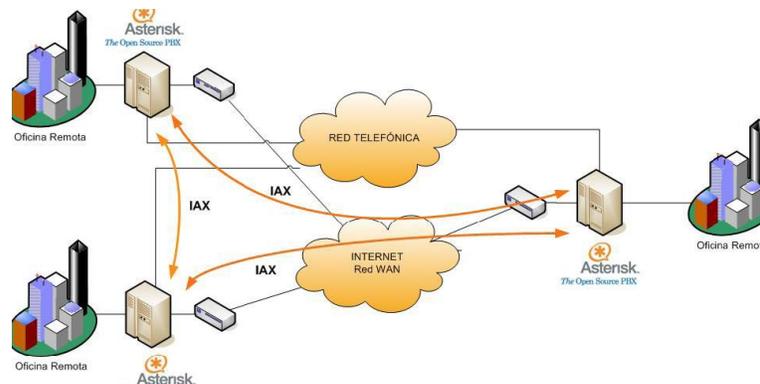


Figura 30: Ejemplo de sistema distribuido

Entre otras de sus funcionalidades están:

- Reducción de costos (Asterisk sólo puede ser comparado a un PBX digital)
- Tener control de su sistema de telefonía
- Ambiente de desarrollo fácil y rápido
- Rico y abundante en recursos
- Es posible proveer contenido dinámico por teléfono

- Plan de discado flexible y poderoso
- Corre bajo Linux y es código abierto

Otra ventaja de Asterisk ha sido el canal SIP que permite conectar cualquier terminal IP, softphone o proveedor mediante este protocolo.

4.4 Arquitectura de Asterisk

4.4.1 Directorios Utilizados por Asterisk

Asterisk usa ciertos archivos en el sistema operativo Linux para gestionar varios aspectos del sistema, tales como grabación de voz, mensajes de voz y configuración de archivos; y son creadas durante la instalación de Asterisk: algunos de estos archivos son configurados en el archivo `asterisk.conf`.

Entre los archivos creados por Asterisk tenemos:

`/etc/asterisk/`

Este directorio contiene los archivos de configuración de Asterisk; sin embargo el archivo `zaptel.conf` es localizado en el directorio `/etc/`. Cuyo archivo se utiliza para configurar el hardware Zaptel que es utilizado por asterisk.

El archivo `zapel.conf` es donde se configura las especificaciones de las interfaces de TDM, requeridas por la tarjeta Zaptel.

Hay dos archivos de configuración con nombre similares, pero son archivos diferentes, estos son:

- `/etc/zaptel.conf`: Configuración de las interfaces del Hardware
- `/etc/asterisk/zapata.conf`: configuración para utilizar las interfaces de hardware

`/usr/lib/asterisk/modules/`

Este directorio contiene todos los módulos cargables de Asterisk. En este directorio están varias aplicaciones, codecs, formatos y canales usados por Asterisk. Por defecto Asterisk carga todos estos módulos en el arranque; para deshabilitar algunos de estos módulos que no se estén utilizando, existe el archivo `modules.conf`, sin embargo hay que recordar que ciertos módulos son requeridos por Asterisk o dependen de otros módulos y tratar de cargar Asterisk sin estos módulos puede causar error en el arranque.

`/var/lib/asterisk`

Este directorio contiene el archivo `astdb` y un número de subdirectorios.

El archivo `astdb` contiene la información local de la base de datos de Asterisk, cuales se asemejan un poco a los archivos de registro de Microsoft Windows.

La base de datos de Asterisk es una simple implementación basada en V1 de la base de datos Berkeley. El origen del archivo `db.c` en Asterisk afirma que esta versión fue elegida por las siguientes razones:

La implementación de DB3 es liberada bajo una alternativa de licencia incompatible con GPL. Por lo tanto, con el fin de mantener la concesión de licencias de Asterisk, fue decidido usar la versión 1 ya que es liberado bajo la licencia BSD.

Los subdirectorios en `/var/lib/asterisk` incluyen:

- ✓ **Agi-bin/:** este directorio contiene los scripts personalizados, que pueden interactuar con Asterisk a través de las diversas aplicaciones AGI.

- ✓ **Firmware/:** este directorio contiene el firmware¹⁹ para diversos dispositivos compatibles de Asterisk. Este comúnmente contiene solo el directorio `iax/`, cual contiene la imagen binaria para Digium's IAXy.

¹⁹ **Firmware:** es un bloque de instrucciones de programa para propósitos específicos, grabado en una memoria tipo ROM

- ✓ **Imágenes/:** aplicación que comunica con los canales soportando imágenes graficas vistas en el directorio *imagenes/* . Muchos canales no soportan la transmisión de imágenes por lo que este directorio se utiliza raramente. Sin embargo, si muchos dispositivos que soportan y hacen uso de imágenes graficas son liberadas, este directorio llegaría hacer más relevante.

- ✓ **Keys/ :** Asterisk puede usar un privado o público sistema de claves para autenticar sus compañeros conectados a través de un cuadro de firma digital RSA. Si se coloca un par de clave pública en el directorio *keys/* los pares pueden ser autenticadas por los canales de apoyo a este método (tales como el canal de IAX2).
Lo códigos privados nunca son distribuidos para el publico.

- ✓ **mohmp3/:** cuando se configura Asterisk para la música en suspenso, las aplicaciones utilizan esta función para ver los archivos MP3 en el directorio *mohmp3/*.

- ✓ **Sounds/:** Todos los disponibles mensajes de voz para Asterisk residen en el directorio *sounds/*. El contenido de las instrucciones básicas incluidas con Asterisk son el archivo *sounds.txt* localizado en el directorio código de origen. El contenido de las indicaciones adicionales se encuentra en el archivo *sounds-extra.txt* para el cual se ha extraído los paquetes *asterisk-sounds*.

/var/spool/asterisk/

El directorio de Asterisk contiene varios subdirectorios, incluyendo *dictate/*, *meetme/*, *monitor/*, *outgoing/*, *system/*, *tmp/*, y *voicemail*.

Asterisk supervisa el directorio saliente de archivos de texto que contienen la información de la palabra solicitada. Estos archivos permiten generar una llamada simple, moviendo la estructura correctamente en el archivo *outgoing/* directory.

Llamando los archivos que se colocan en *outgoing/* directory estos contiene información útil, tales como Contextos, Extensiones y Prioridad donde responde a las llamada al iniciarse o simplifica la aplicación y sus argumentos.

/var/run/

El directorio */var/run/* contiene los identificadores de proceso de información para activar todos los procesos en el sistema, incluyendo Asterisk (como especifico el archivo *asterisk.conf*)

/var/log/asterisk/

Este directorio es donde se encuentran la información de los registros de Asterisk. Se puede controlar el tipo de información, estando validado a varios archivos, editando el archivo *logger.conf*.

/var/log/asterisk/cdr-csv

Este directorio es usado para almacenar los CDRs en formato CSV (separador de valor por comas). Por defecto la información es almacenada en el archivo *Master.csv*, pero individuales cuentas pueden ser almacenadas en sus propios CDRs en separados archivos con el uso de la opción código de cuenta.

Cuadro Resumen De Los Principales Ficheros De Configuración De Asterisk.

ARCHIVO	FUNCION DEL ARCHIVO DE CONFIGURACION
<i>zaptel.conf / zapata.conf</i>	Configuración de líneas y extensiones analógicas y digitales
<i>capi.conf</i>	Configuración de líneas y extensiones RDSI básicas
<i>sip.conf</i>	Configuración de extensiones y conexiones con Operadores IP u otros ASTERISKs
<i>h323.conf / oh323.conf</i>	Configuración de conexiones con Operadores IP
<i>mgcp.conf</i>	Configuración de extensiones MGCP
<i>iax.conf</i>	Configuración de extensiones y conexiones con Operadores IP u otros ASTERISKs
<i>voicemail.conf</i>	Configuración del Sistema de Mensajería
<i>extensions.conf</i>	Configuración del Plan de numeración interno y externo
<i>musiconhold.conf</i>	Configuración del Sistema de música en espera

Tabla 11: Resumen de los principales archivos de Asterisk.

4.4.2 Esquema de La Arquitectura de Asterisk

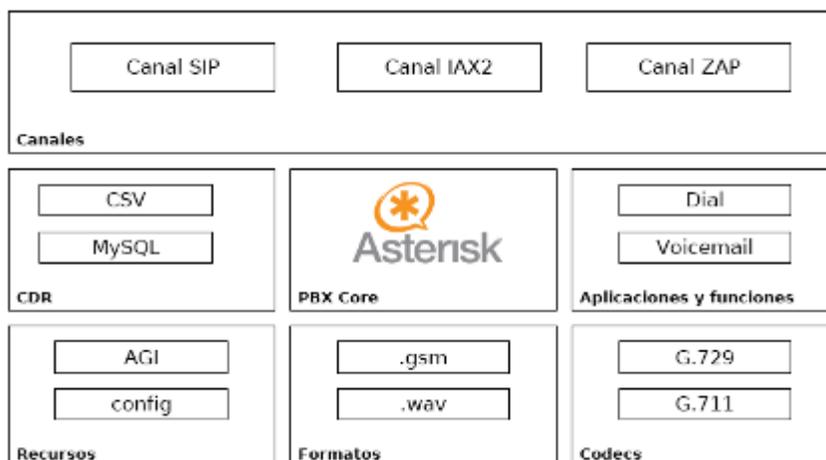


Figura 31: Esquema de la arquitectura de asterisk.

4.4.2.1 Canales

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

En un principio las conexiones de telefonía eran siempre analógicas y por eso, más susceptibles a ruidos y ecos. Más recientemente, buena parte de la telefonía paso para el sistema digital, donde la señal analógica es codificada en forma digital usando normalmente PCM (Pulse Code Modulation). Esto permite que un canal de voz sea codificado en 64 Kilobits/segundo sin ser compactado.

Algunos de los hardwares que Asterisk soporta son:

- Zaptel – Wildcard T410P – Placa E1/T1 con cuatro puertas (PCI 3.3 volts apenas)
- Zaptel – Wildcard T405P – Placa E1/T1 con cuatro puertas (PCI 5 volts apenas)
- Zaptel – TDM400P – Placa con cuatro puertas para tel. analógicos y ADSI,
- Zaptel - TE110P – Placa con E1/T1 con una puerta, medio-comprimido.
- Quicknet, - las placas quicknet, tanto PhoneJack como LineJack pueden ser usadas con Asterisk

- ISDN4Linux – Es un driver antiguo para placas ISDN BRI, acceso básico. Placas de este standard podrán ser usadas en Asterisk.
- ISDN CAPI – Es la otra forma de soportar las placas ISDN BRI en Linux. Placas que soportan este standard podrán ser usadas con Asterisk.
- Voicetronix: poseen placas con mayor densidad de canales FXS y FXO que las placas de Digium.

Canales que Asterisk soporta:

- Agent: Un canal de agente DAC.
- Console: Cliente de consola de Linux, driver para placas de sonido (OSS ou ALSA).
- H323: Uno de los protocolos más antiguos de VoIP, usado en muchas implementaciones.
- IAX e IAX2: Inter-AsteriskExchange Protocol, El protocolo propio de Asterisk.
- MGCP: Media Gateway Control Protocol, otro protocolo de VoIP.
- Modem: Usado para líneas ISDN y en módems.
- NBS: Usado para broadcast de sonido.
- Phone: Canal de telefonía de Linux.
- SIP: Session Initiation Protocol, el protocolo de VoIP más común.
- Skinny: Un driver para el protocolo de los telefonos IP de Cisco.
- VOFR: voz sobre frame-relay de Adtran.
- VPB: Líneas telefónicas para placas de Voicetronix.
- ZAP: Para conectar telefonos y líneas con placas de Digium. También usado para TDMoE (TDM sobre Ethernet) y para Asteriskzphfc (ISDN en modo NT).
- Unicall: Usado para líneas digitales con señalización E1/R2.

4.4.2.2 Codecs

Asterisk Soporta Los Siguietes Codecs.

Codecs de audio

- Sirven para digitalizar la voz humana.

- Cuando dos dispositivos van a establecer una comunicación, acuerdan el codec a utilizar.
- Factores importantes a la hora de seleccionar un codec:
 - Calidad de audio.
 - Coste computacional.
 - Gasto de ancho de banda.
 - Comparación de ancho de banda utilizado por los codecs.

Codecs de vídeo

- En modo passthrough:
 - H.261
 - H.263
 - H.264
- Solo son soportados en modo passthrough²⁰, por lo que no es posible hacer transcoding²¹.
- Disponen de su formato asociado, por lo que se pueden usar para IVRs visuales o video voicemail.

4.4.2.3 Formatos

Asterisk soporta multitud de formatos: WAV, mp3, gsm, h.264, etc. y se necesitan para reproducir ficheros, p.e. en un IVR, Playback, Voicemail,

4.4.2.4 Aplicaciones y funciones

Se han desarrollado muchas aplicaciones de IVR, MultiConferencia, etc, etc. Pudiendo desarrollar todas aquellas aplicaciones más mediante AGI (Asterisk Gateway Interface) pudiendo desarrollarse aplicaciones en C, C++, perl, php, etc.

²⁰ La técnica del "passthrough" permite enviar una consulta SQL a una base de datos sin necesidad de pasar por el administrador de ODBC. La ventaja principal de utilizar "passthrough" es obtener un mejor rendimiento ya que la consulta es procesada por menos componentes.

²¹Transcoding: a la conversión directa (de digital a digital) de un **códec** a otro, en general con pérdida de calidad.

4.4.2.5 Núcleo PBX

Cuando Asterisk es iniciado, el Cargador de Módulos Dinámicos (*DML-Dynamic Module Loader*) carga e inicializa cada uno de los *drivers* de los diferentes canales, los formatos de los archivos, codecs, aplicaciones y más, enlazándolos con las apropiadas

API internas. Esto permite al Núcleo de Conmutación (*Switching Core*) de la PBX Asterisk comenzar a aceptar llamadas desde las interfaces y responderlas acorde con el plan de marcado, usando el Ejecutor de Aplicaciones (*Application Launcher*) el cual ejecuta aplicaciones, las cuales realizan servicios para los usuarios, como llamar a los teléfonos, correo de voz, reproducción de archivos, etc.

4.4.3 Dial Plan

El dialplan, que podría ser traducido literalmente a Plan de marcado, es la parte medular de la central telefónica Asterisk. Esta constituido por una serie de instrucciones que define la manera en que se manejan las llamadas entrantes y salientes del sistema.

Se encuentra ubicado en el directorio */etc/asterisk/* su edición se realiza en el archivo de configuración: ***extensions.conf***

La estructura del dialplan se divide en cuatro elementos: Contexto, extensiones, prioridades y aplicaciones.

Contexto (Context)

Son las secciones en las que se divide el dialplan, se utiliza para ordenar un número determinado de extensiones en grupos diferentes, en donde cada grupo se mantiene aislado del otro, a menos que se le permita a una extensión pasar a un contexto diferente del que fue creado.

Un contexto se declara escribiendo el nombre que se le quiera asignar entre corchete o paréntesis “[()]”, permite letras de la A a la Z en mayúsculas o minúsculas así como números.

Ejemplo:

[incoming]

[globals]

Todas las definiciones realizadas después de la declaración del contexto pertenecen a este, hasta que se encuentre la definición de otro contexto. Asterisk maneja dos contextos especiales [general] y [globals], por lo que se debe evitar utilizar esos nombres.

Contexto Especiales.

[general]

El contexto [general] configura unas pocas opciones generales como son:

- **static** : Indica si se ha de hacer caso a un comando "save dialplan" desde la consola. Por defecto es "yes". Funciona en conjunto con "writeprotect"

- **writeprotect** : Si writeprotect=no y static=yes se permite ejecutar un comando "save dialplan" desde la consola. El valor por defecto es " no".

- **autofallthrough** : Si está activado y una extensión se queda sin cosas que hacer termina la llamada con BUSY, CONGESTIÓN o HANGUP Si no está activada se queda esperando otra extensión. Nunca debería suceder que una extensión se quede sin cosas que hacer como explicaremos posteriormente.

- **clearglobalvars** : Si está activado se liberan las variables globales cuando se recargan las extensiones o se reinicia Asterisk.

- **priorityjumping** : Si tiene valor 'yes', la aplicación soporta 'jumping' o salto a diferentes prioridades. En desuso

En general estas opciones no son muy importantes y se pueden dejar tal y como aparecen por defecto.

[globals]

En este contexto se definen las variables globales que se van a poder utilizar en el resto de los contextos. Por ejemplo

CONSOLE=Console/dsp; indica que cuando hagamos referencia a la variable CONSOLE estamos llamando a /Console/dsp

Las variables suelen ponerse siempre en mayúsculas para diferenciarlas posteriormente.

Cada vez que se crea un canal de comunicación SIP, IAX, H.323 o de telefonía, se debe definir un contexto. De esta manera, se provee de un punto de inicio a la interacción de la central con los dispositivos conectados a dicho canal.

Extensiones (Extensions)

En Asterisk una extensión es la definición de una serie de pasos que serán ejecutados en base a su "prioridad". Se activan de manera automática en respuesta a llamadas entrantes o por dígitos marcados por los usuarios finales; los comandos se ejecutan generalmente en el orden secuencial, pero algunos como Dial() y Gotolf(), tienen la capacidad de re-direccionar a alguna otra parte, basado en ciertas condiciones.

Cuando se marca una extensión, el comando marcado con prioridad de 1 es ejecutado, seguido por el comando con prioridad 2, y así sucesivamente. Esto continúa hasta que:

- la llamada termina (hangu),
- un comando retorna un código de resultado -1 (indicando falla),

- un comando con prioridad más alta no existe (Asterisk “no pasara por alto prioridades faltantes”)
- la llamada se rutea a una nueva extensión.

Una extensión se define de la siguiente manera

*exten => nombre o numero de extensión, **prioridad**, **aplicación** (“parámetros”)*

Ejemplo:

exten => 123,1,Answer()²²

En este caso el nombre de la extensión es 123, con una prioridad de 1, y la acción a realizar el responder la llamada entrante.

Además, al igual que en las PBX tradicionales es posible asignar una extensión específica a un usuario del sistema Asterisk.

Prioridad.

Cada extensión creada en un contexto del dialplan, permite realizar diferentes acciones, esto es posible gracias a que se le puede asignar una prioridad a cada una de las diferentes funciones. Se suelen asignar de forma secuencial, y se realizaran iniciando con la extensión que posea la prioridad de uno (1).

Ejemplo:

exten => 123,1,Answer()

exten => 123,2,Hangup()²³

En este caso, la extensión llamada 123 ejecutara primero la función Answer() de esta manera se iniciara la comunicación, posteriormente, liberara el canal de comunicación creado a través de la función Hangup().

²² *Aplicación del sistema Asterisk que permite responder la llamada entrante.*

²³ *Aplicación del sistema Asterisk libera el canal de conexión con el usuario.*

Prioridad Sin Numeración

Debido a los problemas causados por errores en la asignación de prioridades. A partir de la versión 1.2 de Asterisk, se introdujo el uso de la prioridad “n”, que no es mas que una indicación de next (Siguiete), esto quiere decir que solo será necesario establecer la prioridad uno (1) y posteriormente se asignara “n” como prioridad a las siguientes extensiones, de esta manera un proceso interno de Asterisk tomara el valor anterior e incrementara de uno en uno, así en caso que se requiera asignar una nueva función a la extensión, bastara con ubicarla en el lugar que se desea y asignarle también la prioridad “n”.

Ejemplo:

```
exten => 123,1,Answer()
```

```
exten => 123,n,realizar función
```

```
exten => 123,n,realizar función
```

```
exten => 123,n,ultima function
```

```
exten => 123,n,Hangup()
```

Etiquetas de Prioridad

Asterisk permite etiquetar las prioridades, esto es útil en caso que se desee hacer referencia a una prioridad por algo más que su número. Para etiquetar una prioridad basta con agregar entre paréntesis () el nombre de la etiqueta, justo después de asignar la prioridad.

Ejemplo:

```
exten => 123,n(etiqueta),aplicación()
```

Aplicaciones

Las aplicaciones son las que realizan el verdadero trabajo en el dialplan, se encargan de interactuar con la información recibida en el canal de comunicación, permiten desde reproducir sonidos hasta transferir llamadas en respuesta a los impulsos emitidos por el usuario.

Algunas de ellas como Answer() o Hangup(), se utilizan sin necesidad de brindarles información adicional, Algunas otras, aceptan de uno a varios argumentos separados por comas para realizar diversas acciones.

Extensión “s”

En caso que una llamada se encuentre en el contexto sin hacer referencia a una extensión específica se puede utilizar la extensión “s”, que será la encargada de manejar este canal de comunicación. La letra “s” significa start (inicio).

Ejemplo:

```
[incoming]
exten => s,1,aplicacion()
exten => s,n,aplicacion()
exten => s,n,aplicacion()
```

Aplicaciones del sistema Asterisk

Entre las aplicaciones que permiten una interacción básica entre un canal de comunicación y la central telefónica, se encuentra **Answer()**, que se encarga de contestar un canal que se encuentra sonando.

Otra función de utilidad es **Playback()**, utilizando esta aplicación se podrá transmitir sonidos previamente grabados a través del canal de comunicación, el parámetro que utiliza es el nombre del archivo o la ruta para encontrarlo en caso de solo establecer el nombre, lo buscara en la ruta `/var/lib/asterisk/sounds/`.

La función **Hangup()** libera el canal de comunicación establecido, se recomienda utilizar esta función para asegurarse que la llamada no continua activa en el contexto que se estableció, esta función no utiliza parámetros.

En caso que se desee realizar una acción diferente a la que se encuentra establecida en nuestro contexto o simplemente repetir las extensiones en respuesta

a una opción del usuario, se utiliza la aplicación **Goto()**, a través de ella se podrá enviar la llamada a otro contexto del dialplan, su sintaxis es la siguiente:

```
exten => 123,n,Goto(contexto,extension,prioridad)
```

4.4.4 Función DIAL()

La principal función de asterisk es interconectar llamadas entre diferentes usuarios, y en algunos casos redes con diferentes métodos de comunicación como telefonía análoga y VoIP. La aplicación **Dial()** es la encargada de realizar esta importante función.

Esta función acepta cuatro argumentos, el primero es el destino al que se quiere realizar la llamada, este valor esta formado por la tecnología utilizada para enlazar la llamada que puede ser: Zap para tecnología análoga y canales E1/T1, o SIP e IAX para tecnología VoIP. Seguido del punto destino o recurso que se desee alcanzar. Por ejemplo cuando, cuando se desea llamar a un dispositivo SIP, previamente definido en sip.conf, el destino se escribe **SIP/Usuario1**.

Ejemplos:

Llamada a través del canal SIP:

```
exten => 123,1,Dial(SIP/Usuario1)
```

; Cuando la extensión 123 sea solicitada la llamada se enviara al Usuario1

Esta función además permite concatenar canales, si deseamos que el usuario sea buscado en más de un canal de comunicación, esto se realiza con el signo “&”.

Ejemplo:

```
exten => 123,1,Dial(Zap/1&Zap/2&SIP/Jane)
```

; En este caso se han definido previamente dos canales para telefonía tradicional (Zap/1, Zap/2) además del Usuario SIP. El primero en responder la llamada será quien se quede con el uso del canal.

Al mismo tiempo esta función permite enlazar la llamada con usuarios que no estén definidos previamente.

Sintaxis:

`Dial(tecnologia/usuario[:password]@host_remoto[:puerto][/extencion_remota])`

Por ejemplo la extensión:

```
exten => 500,1,Dial(IAX2/guest@misery.digium.com/s)
```

; Enlazara a un usuario ubicado en los servidores de Digium utilizando el protocolo IAX.

El segundo elemento de la función Dial() es el tiempo en segundos que se deberá esperar antes de abandonar los intentos por alcanzar al destinatario, y pasar a la siguiente prioridad. En caso de no especificarse el tiempo, Dial() continuara hasta que se conteste la llamada o el llamante decida terminar con el intento de conexión.

Ejemplo:

```
exten => 123,1,Dial(Zap/1,10)
```

```
exten => 123,n,Playback(vm-nobodyavail)
```

```
exten => 123,n,Hangup()
```

Se llamara a la extensión 123 a través del canal análogo Zap/1, intentara durante 10 segundos, en caso que nadie responda reproducirá el archivo *vm-nobodyavail*, luego colgara.

El tercer elemento es una cadena de texto que permite modificar el funcionamiento de la función Dial(), existen múltiples opciones entre ellas se encuentra la opción “m” que permite reproducir un sonido a la persona que llama en lugar del típico sonido del teléfono.

```
exten => 123,1,Dial(Zap/1,10,m)
```

El ultimo argumento que se puede utilizar con la función dial(), es una dirección URL, si el destinatario permite recibir una dirección URL en el momento de la llamada la dirección será enviada. Por ejemplo los teléfonos IP permiten esta funcionalidad. Aunque en realidad este argumento sea rara vez utilizado.

Aunque para funcionar la función dial puede funcionar solamente con el primer parámetro (destino), en caso que se desee utilizar solamente uno de los demás argumentos se puede utilizar la siguiente sintaxis:

```
exten => 1,1,Dial(Zap/1,,m)
```

En este caso la función enviara la llamada por el canal Zap/1 y reproducirá el sonido selecciona al momento de timbrar, y no fue necesario especificar un tiempo máximo.

4.4.5 Limitaciones de la Arquitectura de Asterisk

Asterisk usa una CPU de servidor para procesar los canales de voz, en vez de tener un DSP (procesador de señales digitales) dedicado a cada canal. En tanto que esto permitió que el costo fuese reducido para las placas E1/T1, el sistema es muy dependiente de la performance de CPU.

4.5 Asterisk y la Telefonía Tradicional

Para poder inter-operar con la telefonía tradicional, Asterisk necesita hardware específico.

El principal sponsor y desarrollador de hardware para Asterisk es: DIGIUM. Además existen otras tarjetas de comunicaciones clónicas, pero el resultado no es siempre el esperado.

Otros fabricantes suministran tarjetas de calidad tales como:

- Junghans
- Eicon
- Beronet
- Sangoma

Para operar con líneas analógicas, se necesitan tarjetas con interfaces FXO

- Ejemplo: Digium TDM01B

Para operar con teléfonos analógicos o centralitas clásicas, se requieren interfaces FXS

- Ejemplo: Digium TDM10B
-

En ambos casos, el driver a utilizar es zaptel, la configuración se almacena en `/etc/zaptel.conf`

En líneas digitales (RDSI), en Europa existen dos tipos:

- BRI: Acceso básico, proporciona 2 canales de voz.
- PRI: Acceso primario, proporciona 30 canales de Voz (E1).

Asterisk soporta perfectamente ambos tipos de líneas digitales, con hardware específico:

Para primarios, Digium proporciona tarjetas de hasta 4 puertos, siendo zaptel el driver a utilizar.

Para accesos Básicos, Digium no proporciona hardware, si bien existen varios fabricantes con alternativas de calidad:

- Eicon, Junghanns

4.5.1 Asterisk y la Telefonía Tradicional: Líneas Analógicas

Para operar con las tarjetas con interfaces FXS / FXO, Asterisk utiliza el subsistema Zaptel: Zapata Telephony. Para el cual es necesario descargar e instalar el paquete, dicho paquete está disponible en <http://ftp.digium.com/pub/zaptel/zaptel1.2.2.tar.gz>²⁴

Una vez instalado el sistema zaptel, es necesario configurarlo en `/etc/zaptel.conf`

- Definir zonas (para frecuencias de tonos)

²⁴ Pagina publicada el 13-Mar-2008 y fue consultada el 09 de agosto del 2008

- Definir interfaces en los canales: FXS / FXO

Se puede verificar la configuración correcta con el comando instalado: `ztcfg v`

4.5.2 Asterisk y la Telefonía Tradicional: Líneas Analógicas II

Asterisk utiliza los módulos provistos por `zaptel` para acceder al hardware.

Este enlace se configura en: `/etc/asterisk/zapata.conf`

Aspectos importantes a configurar:

- `Context`: contexto donde irán las llamadas generadas por cada canal.
- `echo cancel`: cancelación de echo (problema importante en telefonía).

4.6 VoIP y Asterisk

Asterisk soporta gran variedad de protocolos de comunicación de VoIP y es compatible con la mayor parte de fabricantes del hardware empleados para telefonía IP (teléfonos, adaptadores, router, etc.)

Por sus características Asterisk se presenta como una elección inmejorable para las soluciones de VoIP de las empresas, al integrar de forma sencilla las conexiones telefónicas tradicionales con los nuevos sistemas de voz. Soportando la mayor parte de los protocolos utilizados:

- SIP (Session Initiation Protocol). El principal protocolo de señalización utilizado en el mundo de la VoIP. Estándar abierto y de grandes posibilidades.
- H.323. Primer estándar VoIP, basado en protocolos de la RDSI, ha sido sustituido por el protocolo SIP, aunque todavía sigue encontrándose en aplicaciones de Videoconferencia principalmente.
- IAX (Inter.-Asterisk Exchange). Inicialmente diseñado para la comunicación entre Asterisk remotos, y actualmente empleado también entre servidor y

cliente VoIP. De reciente creación se ha revelado como un protocolo robusto, potente y flexible. Numerosos fabricantes de hardware lo implementan en sus equipos.

- MGCP (Media Gateway Control Protocol). Protocolo de VoIP de arquitectura compleja.
- SCCP (Skinny Client Control Protocol). Protocolo propietario de Cisco para la gestión entre los teléfonos y su servidor de VoIP, "CallManager".

4.7 Utilización de Asterisk a Través de SIP

Cuando se utiliza el protocolo de inicio de sesión SIP para llevar a cabo la señalización de la llamada, es necesario recordar la participación de los elementos finales de comunicación como son el agente usuario y el agente cliente.

La función de asterisk a través de este protocolo es la de "Back-To-Back agente de usuario" debido a que realiza ambas funciones de cliente y servidor; al momento de realizar una llamada la primera conexión que se realiza es entre asterisk y un usuario, posteriormente el sistema inicia una nueva conexión con el dispositivo destino, de esta manera se crea el canal de comunicación entre ambos elementos, la comunicación multimedia fluye a través del servidor y desde el punto de vista de los usuarios ellos se encuentran a nivel de señalización comunicándose solo con Asterisk.

Para desarrollar canales de comunicación SIP entre los diferentes dispositivos de la red, estos deben configurarse en el archivo **sip.conf** ubicado en la ruta **etc/asterisk/**.

Ejemplo:

```
[general]
```

```
[1000]
```

```
type=friend
```

```
context=phones
```

```
host=dynamic
```

Esta sencilla configuración permitirá crear el dispositivo SIP con extensión 1000. Ya que SIP envía mensajes completos similares a HTTP, es posible asignar directamente nombres de usuario que no sean mayores a 255 caracteres.

Se ha definido como tipo “friend” ya que se permitirá la comunicación desde el dispositivo hacia el servidor y viceversa, los otros dos tipos son user y peer. Asterisk reconoce como user un dispositivo que podrá ingresar al dialplan y como peer a son las llamadas que salen vía la aplicación Dial(). El tipo friend abarca ambos.

El campo Host define la ubicación en la red del dispositivo, para que Asterisk pueda direccionar las llamadas hacia el. Este campo permite asignación directa de IP y cuando esta en dynamic recibe un paquete REGISTER de parte del elemento final.

Además es posible utilizar autenticación de usuarios, esto se realiza habilitando la opción “secret”, que asignara un password para validar la conexión con el dispositivo que intente conectarse al servidor.

La asignación de puerto de escucha se realiza con la opción “bindport” por defecto SIP acepta solicitudes entrantes a través del puerto 5060

4.8 Usos y Aplicaciones de Asterisk

4.8.1 Usos

Entre las formas que es utilizado Asterisk se encuentran:

- Centralita Nueva con conexiones y extensiones de varios tipos

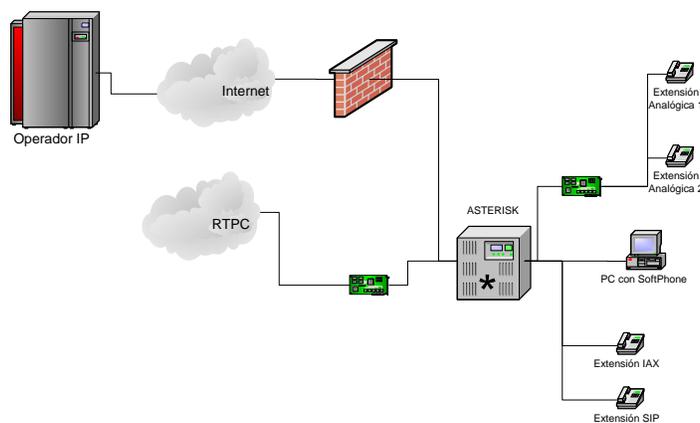


Figura 32: Centralita Nueva con conexiones y extensiones de varios tipos

- Pasarela para dotar a una centralita tradicional de servicios nuevos

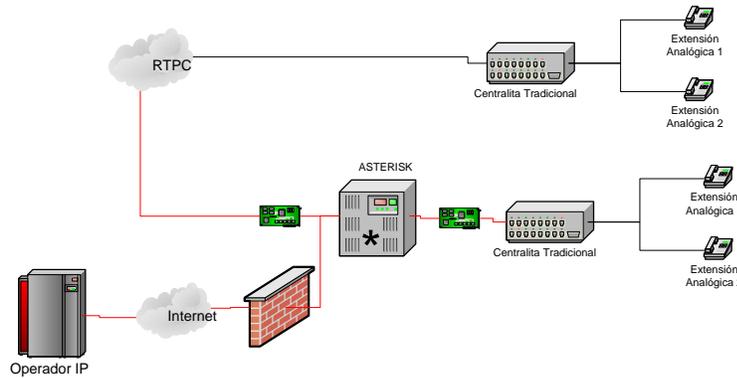


Figura 33: Pasarela para dotar a una centralita tradicional de servicios nuevos

- Pasarela para dotar a una centralita tradicional de nuevas extensiones

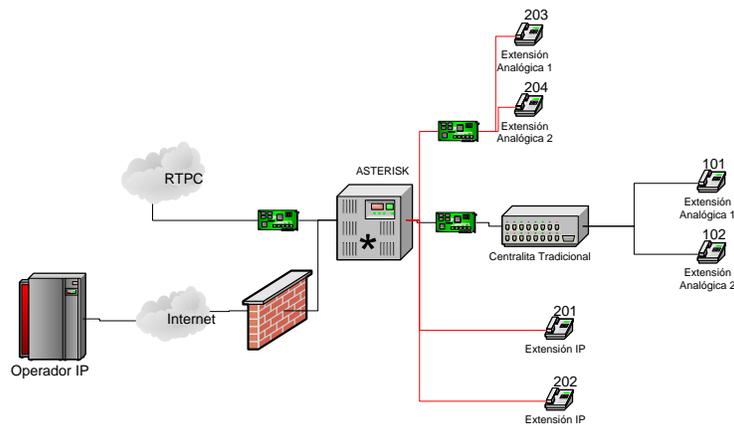


Figura 34: Pasarela para dotar a una centralita tradicional de nuevas extensiones

- Varias Oficinas con un Asterisk Centralizado

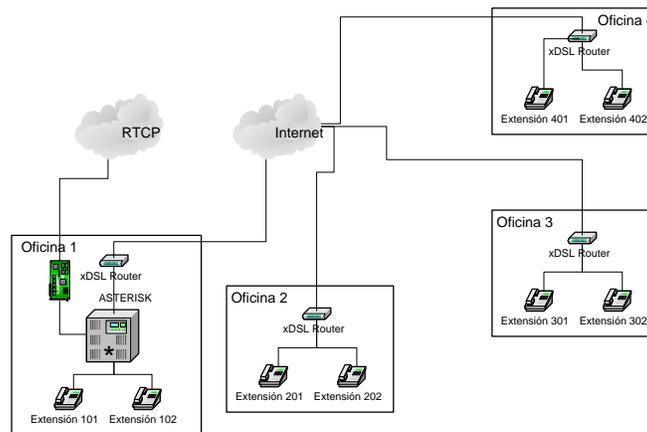
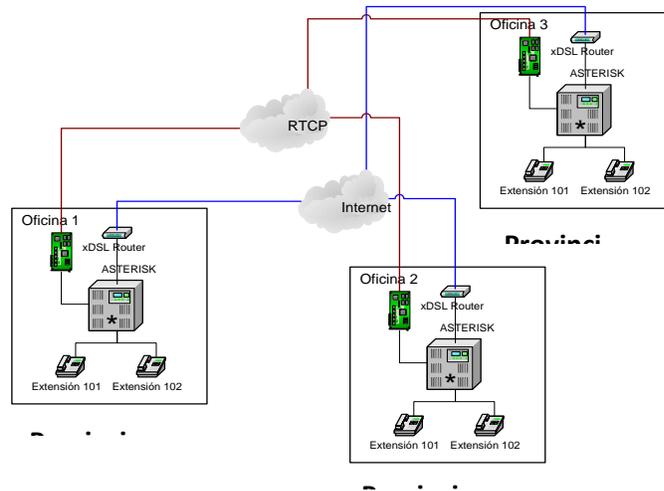


Figura 35: Varias Oficinas con un Asterisk Centralizado

De esta forma, además de comunicaciones entre Sedes, todas las sedes pueden hacer llamadas al exterior a través de la central Asterisk. Las extensiones externas podrán seguir el protocolo SIP o el Protocolo IAX.

- Varias Oficinas con Sistemas Asterisk interconectados



▪ *Figura 36: Varias Oficinas con Sistemas Asterisk interconectados*

Todos los Asterisk Pueden utilizar los otros Asterisk para hablar con números de las provincias donde se encuentran. Entre ellos se pueden conectar mediante IAX o SIP, e incluso se puede montar un plan de numeración único entre todos.

4.8.2 Aplicaciones

- Grupos de Timbrado, Colas y Agentes.
- Llamadas supervisadas.
- Programación con AGI y AMI.
- Panel de administración básico en PHP.
- Aplicaciones en un Call Center: IVR, integración de Asterisk con Software de call center, llamadas predictivas, implementación de campañas.
- Demostraciones.

CAPITULO V

**“GUÍA DE IMPLEMENTACIÓN DE UN
SERVIDOR ASTERISK COMO UNA
SOLUCIÓN DE VOZ SOBRE IP”**

5.1 Elementos a Utilizar

Para el funcionamiento de la Red se necesita los siguientes elementos:

- Teléfono análogo
- Teléfono IP marca POLYCOM IP 301 SIP
- Softphone X-Lite
- Adaptador de teléfono análogo (ATA) *VoIP Solution MTA V 102*
- Modem CSU/DSU marca 2wire
- Cables recto Ethernet Rj45
- Cables Modem Rj11
- Servidor Asterisk

5.2 Requerimientos Mínimos de Hardware y Software.

Para la previa instalación de Asterisk, es necesario contar con una computadora que posea los siguientes requerimientos mínimos para poder ser instalado²⁵.

- Procesador a 500MHz (Pentium3) con 128 MB en RAM
- 5GB en disco duro como mínimo.

Recomendados

- Procesador a 1.5 GHz (Pentium 4)
- 256 MB en RAM
- 10 GB en disco duro.

Utilizados en la guía.

Para el desarrollo de esta guía se utilizara un servidor dedicado, que cuenta con las siguientes características.

- Procesador Pentium D de 2.8GHz
- 1 Gb de memoria Ram
- 20 Gb de disco Duro

Nota: Las pruebas se pueden realizar en una máquina virtual configurada con las características recomendadas de hardware.

²⁵ <http://www.scribd.com/doc/2073022/Guia1-Instalacion-de-Asteriskspco> consultados el 30 de agosto del 2008.

5.3 Desarrollo de La Guía.

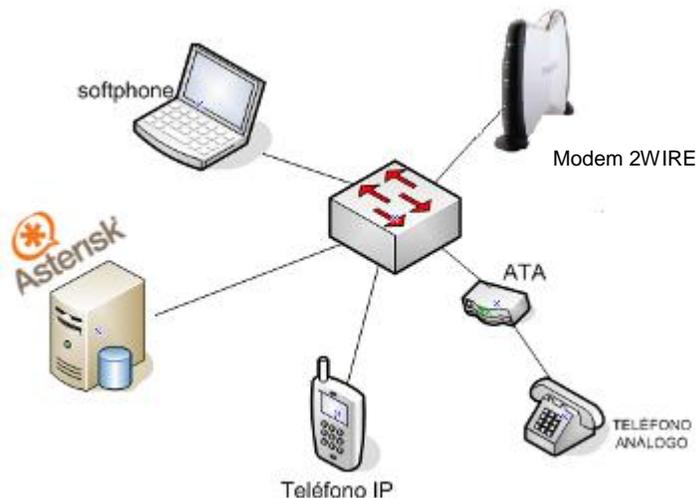


Figura 37: Diagrama y elementos de la Red a implementar.

Las extensiones que se asignara a cada dispositivo serán las siguientes:

- Teléfono IP Ext. 7000
- Telefono Analogo Ext. 8000
- Softphone X-Lite Ext. 9000

PASO 1. INSTALACIÓN DE ASTERISK

El software que se utilizara para la implementación de la PBX Asterisk será AsteriskNow. Se eligió este software ya que es una distribución ligera que incluye el mínimo software imprescindible para montar un servidor Asterisk dedicado. Además permite una instalación amigable debido a su entorno grafico.

AsteriskNow; es un software desarrollado por Digium, se trata de una distribución de Linux especialmente adaptada para hacer funcionar Asterisk en cuestión de minutos ya que viene con todos los requerimientos y dependencias de software preconfigurados y permite al usuario la administración y mantenimiento del servidor de una manera realmente sencilla.

El primer paso para la instalación es obtener una versión de Asterisk Now, donde puede ser descargado de la web en: <http://www.asterisknow.org/>²⁶;

Una vez descargado de la Web la imagen ISO, se procede a quemar la imagen a un CD y luego arrancar la PC.

Advertencia: Cada vez que se ejecuta la instalación se formatea el disco duro y se perderán todos los archivos contenidos en el mismo.

Con el CD introducido saldrá una primera pantalla.

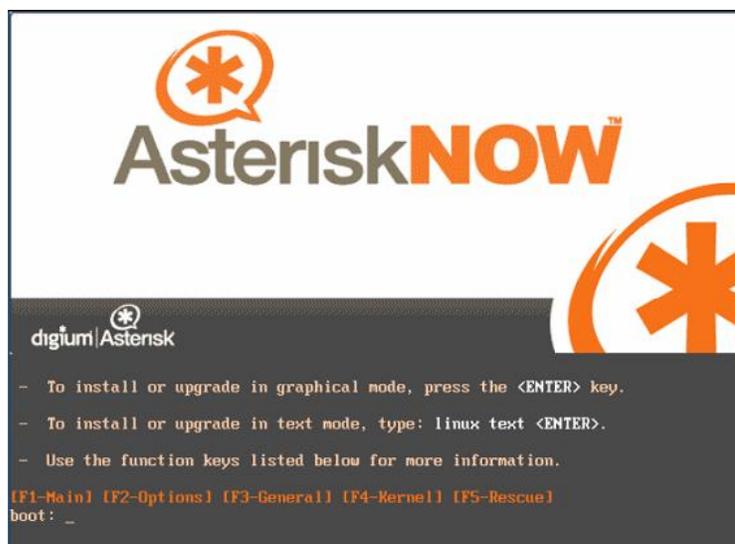


Figura 38: Primera pantalla de arranque de AsteriskNow

Donde se debe dar enter para hacer la instalación en modo grafico; lo que permitirá que salga una pantalla dando la bienvenida a la instalación.

²⁶ <http://www.asterisknow.org/>: ultima fecha de consulta 30 de agosto del 2008



Figura 39: Pantalla de bienvenida de AsteriskNow

En esta ventana se debe pulsar el botón Next. y saldrá la siguiente pantalla, donde se seleccionara la opción de Express installation y pulsar Next.

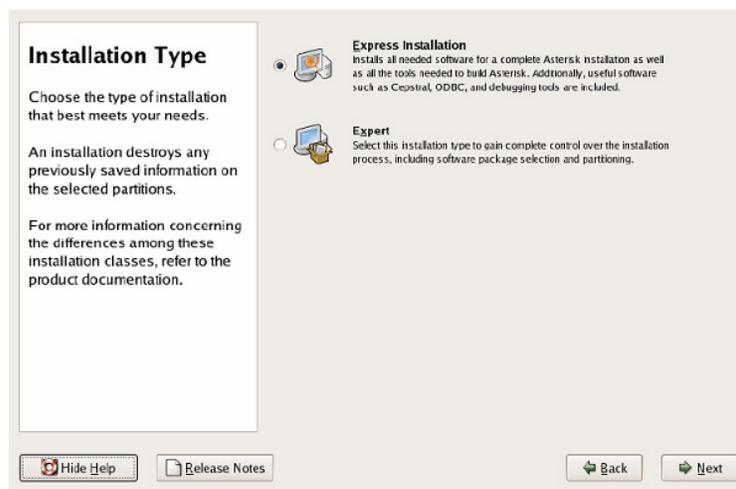


Figura 40: Pantalla de selección de tipo de instalación.

La siguiente pantalla puede variar. Si en el disco donde se desea realizar la instalación de AsteriskNow, existe la instalación de un Windows, saldrá un mensaje indicando que se borrarán todos los datos. Donde se pulsara **yes**.

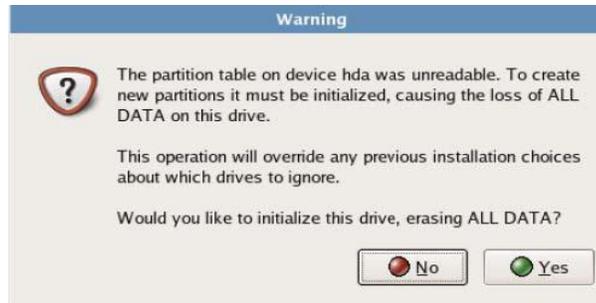


Figura 41: Pantalla de advertencia sobre la perdida de particiones.

Ahora empezara el proceso de partición automático. Donde si solo existen particiones de Linux se seleccionara la primera opción, la segunda opción será seleccionada si se desea eliminar toda las particiones existentes del disco. Una vez seleccionada la opción pulsar next.

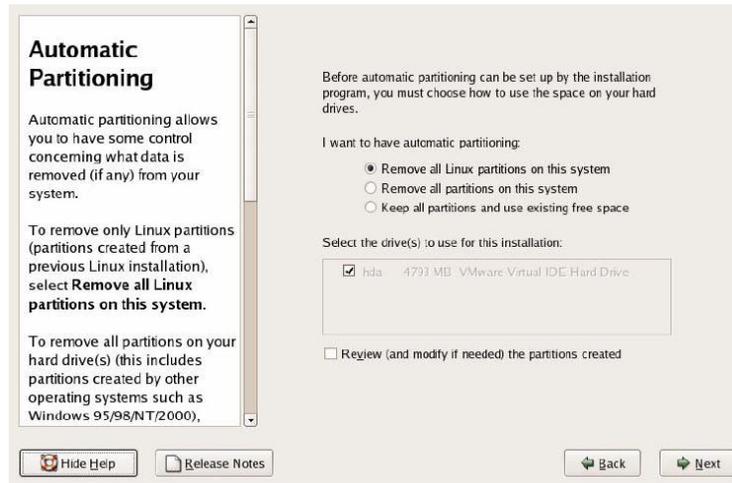


Figura 42: Pantalla de partición automática.

Antes de empezar el proceso de instalación, saldrá un mensaje indicando que si estas seguro de eliminar las particiones. Se pulsara yes.



Figura 43: Mensaje de Advertencia que se eliminaran las particiones existentes.

La siguiente pantalla es para configurar la red del servidor Asterisk. Donde se puede configurar mediante DHCP (obtención automática de datos) o manualmente.

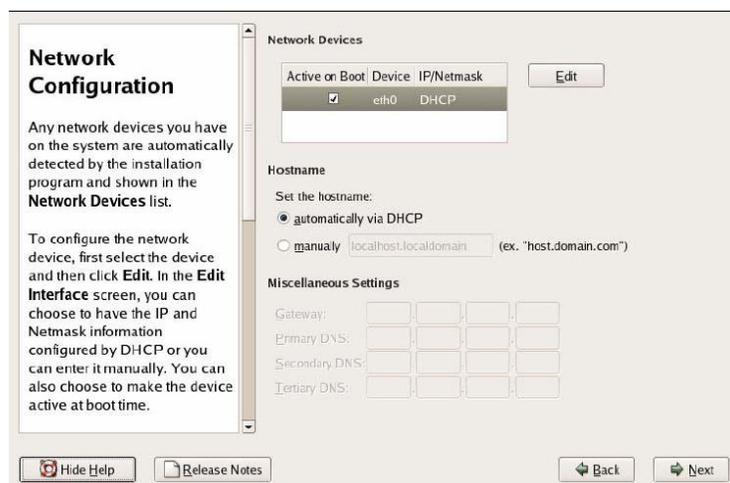


Figura 44: Configuración de la Red.

La siguiente pantalla es para configurar la zona de tiempo.



Figura 45 Configuración de la zona horaria.

Por defecto se crea un usuario denominado ADMIN, donde se solicita la contraseña. Este será el administrador de la parte del asterisk, pero también existe el usuario ROOT que es el administrador de todo, ingresamos la contraseña y pulsamos next.



Figura 46: Pantalla de Ingreso de Password del usuario Admin.

Antes de comenzar la instalación, se muestra una pantalla que indica procesos que se llevan a cabo en la instalación. Pulsar **next**.



Figura 47: Resumen de los procesos que se llevaran acabo en la instalación.

Antes de realizar la instalación, se realiza el formateo del sistema.



Figura 48: Formateo del sistema

Luego comienza el proceso de instalación, saldrán las siguientes pantallas.



Figura 49: Inicio de la Instalación de Asterisk.



Figura 50: finalización de la Instalación de Asterisk.

Una vez finalizada la instalación, solicitará el reinicio.



Figura 51: Solicitud de reinicio del equipo.

Primer Arranque Del Asterisk Now

Una vez finalizada la instalación, se ha reiniciado la PC, donde mostrará una pantalla que solicitará el núcleo de arranque.

La pantalla siguiente saldrá en el primer arranque de Asterisk Now, donde indica que existe un usuario llamado admin., y que su contraseña la que se introdujo durante el proceso de instalación.



Figura 52: Pantalla de Bienvenida.

Por ultimo, la PC se quedara con la consola de Asterisk Now. Donde si se desea se puede actualizar el sistema. El proceso de actualización será de varios minutos.

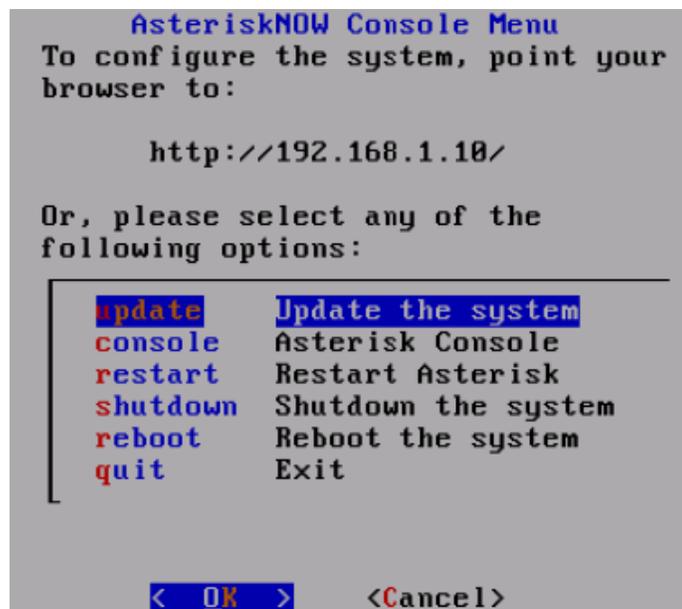


Figura 53: Pantalla principal de AsteriskNow.

Administración de AsteriskNow a Través De Un Navegador

Para administrar Asterisk a través de un navegador se debe tener un ordenador, con un navegador Web, en el cual se debe introducir la IP del servidor Asterisk.

Una vez la pagina cargue se debe de ingresar el usuario ADMIN con su correspondiente contraseña.

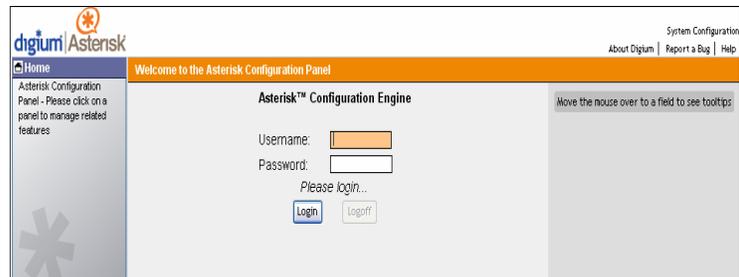


Figura 54: Asterisk a través del navegador.

Una vez el administrador ingresa las claves carga la siguiente pantalla que permite la configuración del servidor de una manera grafica, sin embargo la configuración de Asterisk en este documento se realizara vía consola.

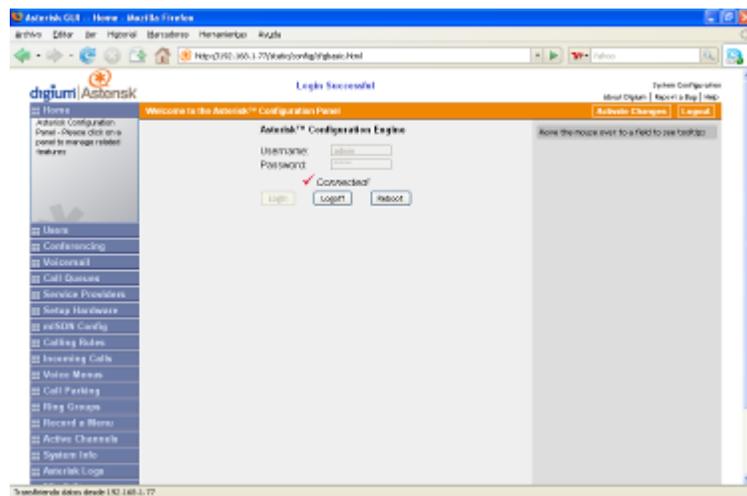


Figura 55: Pantalla principal de la administración de Asterisk vía Web

PASO 2. CONFIGURACIÓN DE ASTERISK VÍA CONSOLA

Para administrar Asterisk vía consola, se necesita un software que permita la comunicación vía SSH, tales como: SecureCRT o Putty.

En este caso se utilizara PuttY, donde se ingresara la IP del servidor para su administración, se selecciona SSH como tipo de conexión y el puerto 22.

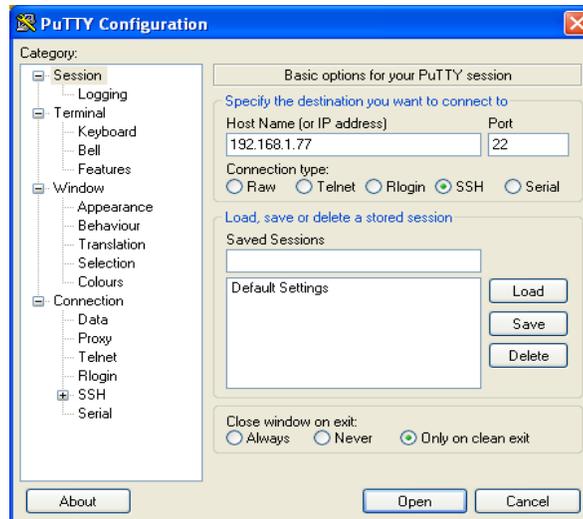


Figura 56: Configuración de Putty para ingresar vía consola al servidor Asterisk

Una vez se ingrese al servidor este solicitara el usuario y contraseña, donde se deberán ingresar.



Figura 57: pantalla de registro de usuario del servidor Asterisk

Una ves el usuario se encuentre dentro de la configuración de asterisk, se debe ingresar al directorio /etc/asterisk

Donde se debe abrir el archivo sip.conf como root para agregar los usuarios además de las características del canal SIP.

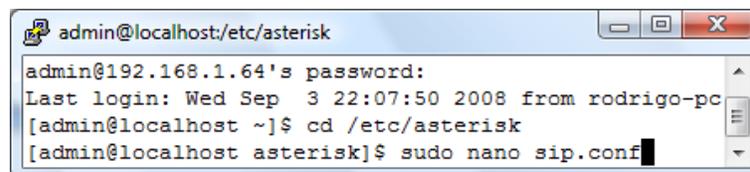
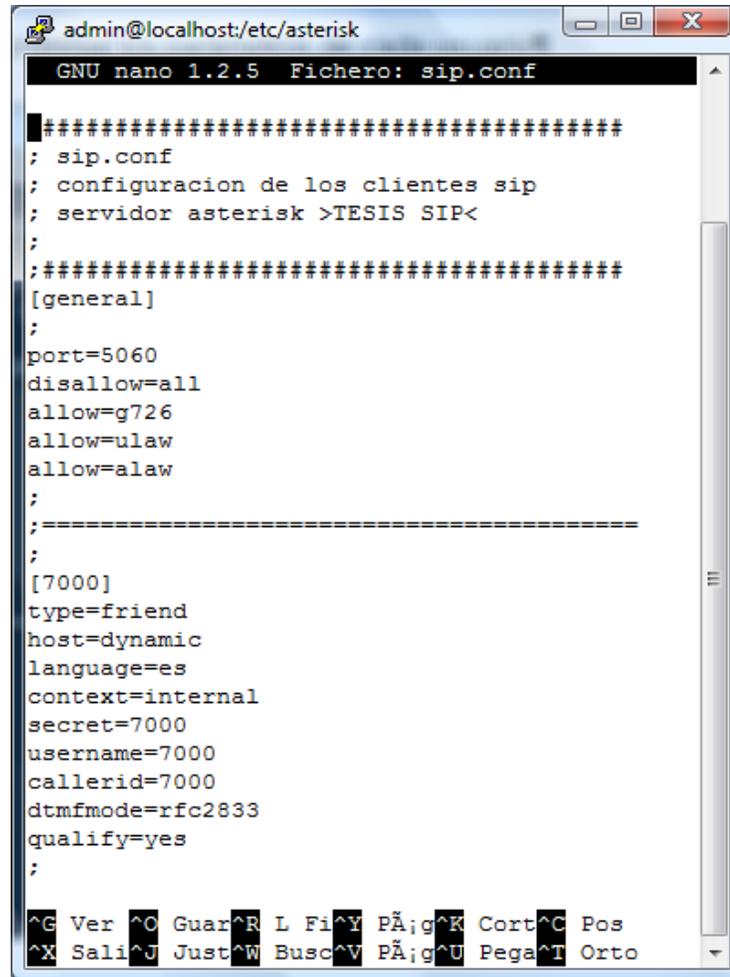


Figura 58: Ingreso al editor nano para editar el archivo sip.conf

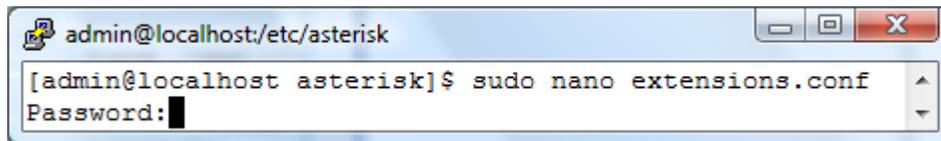
Una vez dentro del archivo se procede a ingresar los parámetros de cada usuario



```
admin@localhost:/etc/asterisk
GNU nano 1.2.5 Fichero: sip.conf
#####
; sip.conf
; configuracion de los clientes sip
; servidor asterisk >TESIS SIP<
;
;#####
[general]
;
port=5060
disallow=all
allow=g726
allow=ulaw
allow=alaw
;
;=====
;
[7000]
type=friend
host=dynamic
language=es
context=internal
secret=7000
username=7000
callerid=7000
dtmfmode=rfc2833
qualify=yes
;
^G Ver ^O Guar ^R L Fi ^Y PÃ;g ^K Cort ^C Pos
^X Sali ^J Just ^W Busc ^V PÃ;g ^U Pega ^T Orto
```

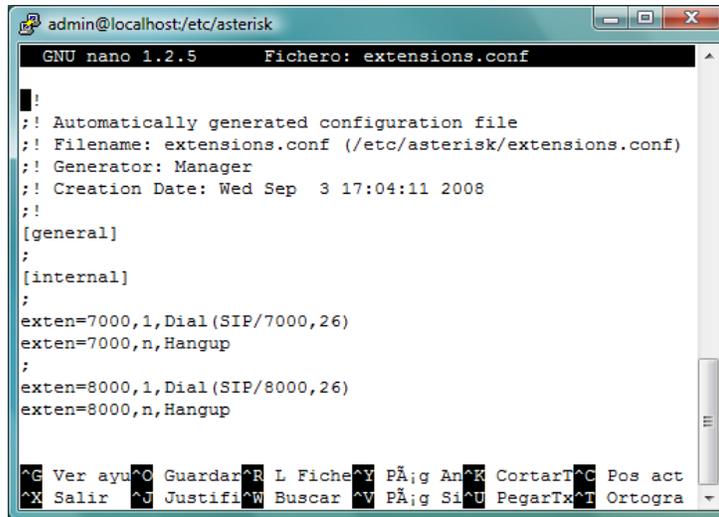
Figura 59: Configuración del archivo sip.conf

Además por cada extensión creada se debe de habilitar la comunicación entre los usuarios modificando el dial plan, que se encuentra en el archivo extensions.conf. Al igual que en el paso anterior se utilizara el editor de texto nano y siempre anteponiendo root al comando nano.



```
admin@localhost:/etc/asterisk
[admin@localhost asterisk]$ sudo nano extensions.conf
Password: █
```

Figura 60: Ingreso al archivo extensions.conf



```
admin@localhost:/etc/asterisk
GNU nano 1.2.5 Fichero: extensions.conf
#!
;!-- Automatically generated configuration file
;!-- Filename: extensions.conf (/etc/asterisk/extensions.conf)
;!-- Generator: Manager
;!-- Creation Date: Wed Sep 3 17:04:11 2008
;!--
[general]
;
[internal]
;
exten=7000,1,Dial(SIP/7000,26)
exten=7000,n,Hangup
;
exten=8000,1,Dial(SIP/8000,26)
exten=8000,n,Hangup
^G Ver ayu^O Guardar^R L Fiche^Y PÄ;g An^K CortarT^C Pos act
^X Salir ^J Justifi^W Buscar ^V PÄ;g Si^U PegarTx^T Ortogra
```

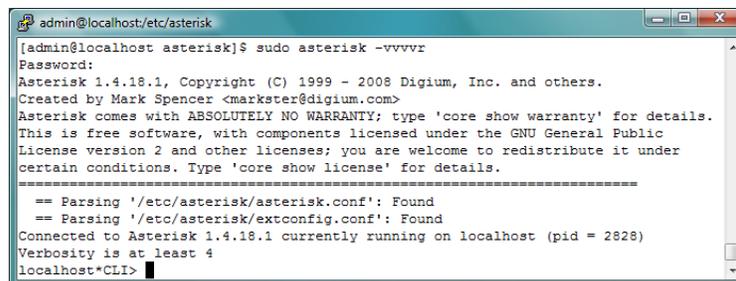
Figura 61: Configuración del archivo extensions.conf

Únicamente se agrega el contexto [internal] que fue definido para cada usuario.

Con este contexto se define la acción a realizar por la PBX al momento que los usuarios marquen los números de las extensiones.

Al haber terminado la configuración esta se aplicara al reiniciar el servidor siempre como root: **sudo /etc/init.d/asterisk restart**

Además es útil monitorear las acciones que realiza el servidor, estado de los usuarios detalles de llamada etc. Esto es posible con el comando **sudo asterisk -vvvvv**



```
admin@localhost:/etc/asterisk
[admin@localhost asterisk]$ sudo asterisk -vvvvv
Password:
Asterisk 1.4.18.1, Copyright (C) 1999 - 2008 Digium, Inc. and others.
Created by Mark Spencer <markster@digium.com>
Asterisk comes with ABSOLUTELY NO WARRANTY; type 'core show warranty' for details.
This is free software, with components licensed under the GNU General Public
License version 2 and other licenses; you are welcome to redistribute it under
certain conditions. Type 'core show license' for details.
=====
== Parsing '/etc/asterisk/asterisk.conf': Found
== Parsing '/etc/asterisk/extconfig.conf': Found
Connected to Asterisk 1.4.18.1 currently running on localhost (pid = 2828)
Verbosity is at least 4
localhost*CLI>
```

Figura 62: Monitoreo de las acciones del servidor

Una vez dentro podemos verificar el estado de los usuarios creados digitando **sip show peers**, que muestra una pantalla con información general de los usuarios creados además de su estado actual:



```
admin@localhost/etc/asterisk
localhost*CLI> sip show peers
Name/username      Host           Dyn Nat ACL Port   Status
-----
8000/8000          192.168.1.72  D      5060  OK (19 ms)
7000/7000          192.168.1.64  D      5060  OK (72 ms)
2 sip peers [Monitored: 2 online, 0 offline Unmonitored: 0 online, 0 offline]
localhost*CLI>
```

Figura 63: Verificación del estado de los usuarios.

Además si habilitamos el debug del servidor con el comando **sip set debug** se puede observar el estado de las consultas realizadas al servidor en tiempo real.



```
admin@localhost/etc/asterisk
localhost*CLI> sip set debug
SIP Debugging enabled
Reliably Transmitting (no NAT) to 192.168.1.72:5060:
OPTIONS sip:8000@192.168.1.72 SIP/2.0
Via: SIP/2.0/UDP 192.168.1.64:5060;branch=z9hG4bK10d93063;rport
From: "asterisk" <sip:asterisk@192.168.1.64>;tag=as7152edea
To: <sip:8000@192.168.1.72>
Contact: <sip:asterisk@192.168.1.64>
Call-ID: 2459113e01ea6c7f6ala420051576a79@192.168.1.64
CSeq: 102 OPTIONS
User-Agent: Asterisk PBX
Max-Forwards: 70
Date: Thu, 04 Sep 2008 05:53:05 GMT
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, SUBSCRIBE, NOTIFY
Supported: replaces
Content-Length: 0
localhost*CLI>
```

Figura 64: Activación del Debug.

Para detener el debug se utiliza el comando **sip set debug off**.

PASO 3. CONFIGURACION DE LOS CLIENTES SIP

Configuracion de Telefono IP Polycom IP 301 SIP

Para configurar el telefono IP a través de la web, se debe ingresar en el navegador la dirección IP del teléfono, ya haya sido esta asignada estáticamente o por DHCP.. Para ingresar a la configuración SIP este solicitará el usuario y password los cuales son por defecto: usuario: *Polycom* password: *456*

En la pantalla que se despliega se configura los parámetros del servidor Asterisk, tales como: la IP, el puerto del protocolo SIP, que por defecto es el 5060.

Servers	
Outbound Proxy	
Address	192.168.1.64
Port	5060
DNS Lookup	DNSnaptr
Server 1	
Address	192.168.1.64
Port	5060
DNS Lookup	DNSnaptr
Expires	
Register	1
Retry Time Out	0
Retry Max Count	0
Line Seize Time Out	30

Figura 65: Ingreso de la IP del servidor al Teléfono IP.

En la sección Line, se configura la extensión y nombre que se le asignara al teléfono IP, esta debe coincidir con los datos ingresados en el archivo sip.conf del servido, en este caso se utilizo la extensión 7000; además se ingresa el password de autenticación que debe de coincidir con el ingresado en el servidor asterisk.

Line 1	
Identification	
Display Name	Rodrigo Castro
Address	7000
Auth User ID	7000
Auth Password	*****
Label	
Type	<input checked="" type="radio"/> Private <input type="radio"/> Shared
Third Party Name	
Num Line Keys	
Calls Per Line Key	
ACD Login Logout	0
ACD Agent Available	0
Ring Type	6

Figura 66: Configuración de extensión y autenticación del teléfono.

Para configurar los códec y la prioridad para ser utilizados, hay que ingresar a Códec Preferences y seleccionar el orden.

Codec Preferences	
G.711u	First
G.711A	Second
G.729AB	Third

G.711u Codec Profile	
Payload Size	20
Jitter Buffer Minimum	40
Jitter Buffer Shrink	500
Jitter Buffer Maximum	160

G.711A Codec Profile	
Payload Size	20
Jitter Buffer Minimum	40
Jitter Buffer Shrink	500
Jitter Buffer Maximum	160

G.729AB Codec Profile	
Payload Size	20
Jitter Buffer Minimum	40
Jitter Buffer Shrink	500
Jitter Buffer Maximum	160

top

Figura 67: Configuración de códec.

Configuración del dispositivo adaptador analogo VoIP Solution MTA V 102

El adaptador analogo utilizado en el desarrollo de esta guía es un ATA VoIP Solution. Para la configuración de este dispositivo, se necesita ingresar al navegador la IP asignada de forma estática o dinámica, la cual solicitará el usuario y password. Estos por defecto son: usuario **root** y password **opensesame**.

Welcome to VoIP TA Web Setting Page.
Please input Login-ID and Password.

Login-ID	root
Password	*****

Figura 68: Pantalla de ingreso a la configuración de adaptador

En SIP Configuration se ingresará en el campo proxy Server IP, la IP del servidor Asterisk y el puerto de este. En Register Server IP debe ir el nombre de dominio de

servidor Proxy, y si no se utiliza un nombre de dominio se ingresara la IP del server proxy. Tambien se ingresa el password de autentificacion configurado tambien en el servidor, y el numero de la extension que en este caso es la 8000.

Proxy Server IP (or Domain)	192.168.1.64	Default
Secondary Proxy Server IP	0.0.0.0	Default
Proxy Server Port	5060	Default
<input type="checkbox"/> Receive an authorized call only		
<input type="checkbox"/> Use outbound proxy		
<input type="checkbox"/> Use outbound proxy within dialog		
Primary outbound proxy	0.0.0.0	Default
Secondary outbound proxy	0.0.0.0	Default
Outbound Proxy Server Port	5060	Default
<input type="checkbox"/> Use proxy as registrar		
Register Server IP (or Domain)	192.168.1.64	Default
Secondary Register Server IP	0.0.0.0	Default
Register Server Port	5060	Default
Register Interval[0,30-2**32-1]	600	sec Default
Retry Interval[10-255]	10	sec Default
Session Expires[0,30-2**32-1]	0	sec Default
Local SIP Port	5060	Default
Local RTP Start Port	5080	Default
User ID 1	8000	Default
Password 1		(write only)
IP Telephone Number 1	8000	Default
Displayname 1	Prueba	Default
User ID 2	UNUSED	Default
Password 2		(write only)
IP Telephone Number 2	UNUSED	Default
Displayname 2	UNUSED	Default
<input checked="" type="checkbox"/> Synchronize Date and Time		

Figura 69: Configuración de SIP Configuración.

En la opción Provisioning Server se elige el tipo de provisión entre HTTP o TFTP, por defecto es usado HTTP. Se debe ingresar también la IP del servidor Asterisk.

Provisioning Server

Provision Type	HTTP	
Provision Server	192.168.1.64	
Retry Interval	8	second
Provisioning Interval	5	minute
Authentication Key	*****	(write only)

Figura 70: Configuración de Provisioning Server

Instalación De Softphone X-Lite

El softphone que se utilizara para realizar las llamadas será es X-Lite, los pasos para instalarlo son los siguientes.



Figura 71: Pantalla de Instalación de X-Lite

Luego se presentara la pantalla de bienvenida, se dará click en next para continuar con la instalación.

Se aceptan los términos de licencia, y finaliza la instalación.

Configuración Del Softphone

Una vez se instalado el sofphone este se ejecutara y se procederá a la configuración.

Dando click derecho sobre la flecha dirigida hacia abajo que se muestra en la pantalla del Softphone se configurara la cuenta de usuario SIP.



Figura 72. Configuración del Softphone

Luego se dará click en la opción SIP Account Setting y se desplegará la siguiente pantalla.

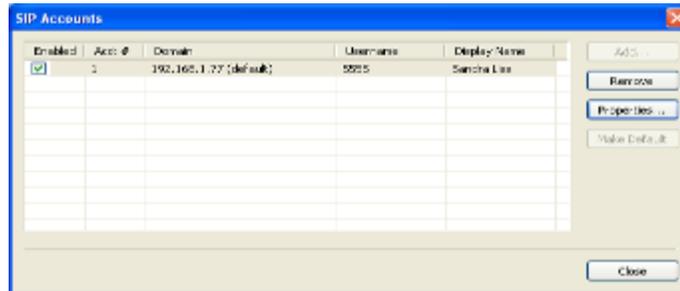


Figura 73: configuración de la cuenta de SIP

En la pantalla anterior se debe presionar el botón Add para agregar un nuevo usuario o el botón Properties para realizar la configuración de la extensión que se le asignará al softphone, en este caso será 9000.

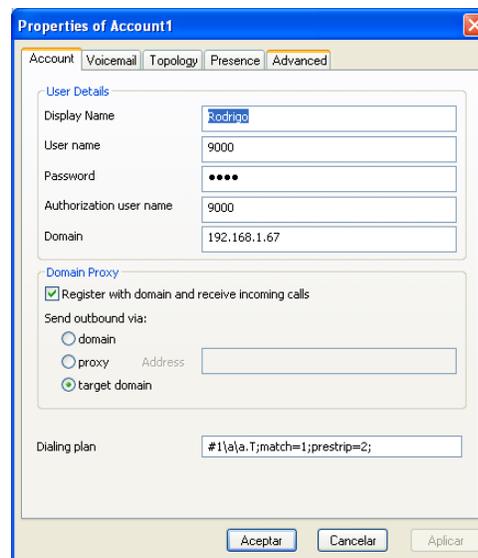


Figura 74: Configuración de los parámetros del Softphone

En dicha pantalla se debe ingresar el nombre que desplegará el softphone, el número de extensión, el password de autenticación que coincidirá con el configurado en el servidor Asterisk y la IP del servidor Asterisk.

Una vez se haya establecido los parámetros se presionará el botón aceptar, y listo el softphone se encontrará ya configurado y mostrará en su pantalla principal el nombre y número de la extensión.



Figura 75: Softphone Muestra la extensión asignada.

5.4 Establecimiento de La Llamada Asterisk

Para verificar la correcta configuración de los usuarios y el servidor se realizan diferentes pruebas de llamada entre los distintos usuarios marcando la extensión correspondiente.

Ejemplo: Establecimiento de llamada entre Softphone X-Lite (Ext. 9000) y teléfono IP Polycom (Ext. 7000).

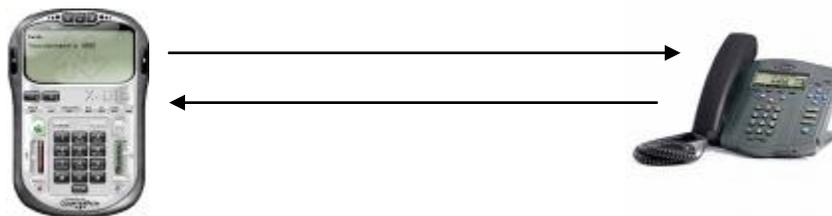


Figura 76: Establecimiento de la llamada entre Softphone y Teléfono IP.

Se observara que la llamada se estable sin dificultad alguna; para seguir verificando el funcionamiento de Asterisk, debe continuarse con las pruebas de llamadas entre los demás dispositivos.

Conclusiones

- La tecnología VoIP sufre aún en la actualidad de muchas limitantes respecto a las telefonías tradicionales; limitantes que no solo son propias de esta tecnología, sino más bien, son causadas por otros factores externos como la disponibilidad de red, poco ancho de banda disponible, o equipos con capacidad de procesamiento insuficiente. Pero el constante desarrollo tecnológico permitirá a corto plazo que el uso de VoIP reemplace a los sistemas de telefonía actuales.
- En la comunicación de voz, los protocolos de señalización realizan las labores más importantes ya que se dedican a la gestión de los recursos de la red; por esta razón es importante seleccionar un buen protocolo de señalización, debido a que estos definen además la estructura de la red a implementar; así como también la eficiencia de la misma.
- El modelo de funcionamiento cliente-servidor del protocolo de inicio de sesión SIP, es una de las razones que le ha permitido ser el protocolo de señalización más utilizado actualmente. Ya que le permite ser eficiente y simple, debido a que su implementación puede ser amplia y fácilmente compatible con diferentes arquitecturas de red; al mismo tiempo facilita el desarrollo de nuevas aplicaciones VoIP.
- El servidor SIP presenta ciertas dificultades al interactuar con PAT (NAT overload) ya que PAT permite la traducción de muchas direcciones privadas internas con una sola dirección pública externa utilizando puertos, y SIP al utilizar el protocolo RTP, encargado de llevar las conversaciones de voz, se genera un conflicto con los puertos ya que RTP utiliza puertos aleatorios para el envío de la señalización, la cual el router no puede saber por cual puerto puede mandar nuevamente la señalización y esto genera que el audio no fluya

correctamente entre los interlocutores; sin embargo este inconveniente puede ser solventado de varias maneras; siendo una de ellas, el uso de un servidor STUN

- Asterisk es un potente central telefónica muy flexible de utilizar en diferentes entornos VoIP, ya que sus archivos de configuración pueden ser modificados por el usuario según sean las necesidades y entorno en el que se desarrollara; sin embargo esto requiere de un administrador que posea conocimientos sólidos sobre el funcionamiento y configuración de Asterisk.
- La interconexión y la administración de los canales de comunicación creados en Asterisk, son administrados a través del Dial Plan, esto hace necesario conocer su estructura y funcionalidades si se desea sacar un mejor provecho a la central telefónica.
- Asterisk está preparado para cubrir prácticamente todas las necesidades de Comunicaciones existentes y lo más importante, es siempre actualizable para nuevos servicios que se vayan añadiendo. La Videoconferencia, el envío de SMS mediante la línea Fija. Servicios Automáticos de respuesta interactiva y demás aplicaciones pueden cubrir prácticamente todas las necesidades existentes en el sector de la Comunicaciones.

Recomendaciones

- El software libre Asterisk cuenta con varias funcionalidades, se recomienda desarrollar un proyecto de implementación a mayor escala sobre el servidor Asterisk para su uso en redes de comunicación más compleja, y lograr así extraer y aplicar todas las ventajas que brinda esta potente herramienta.
- La tecnología de VoIP ya es aplicada por en algunas empresas nacionales e internacionales, por esa razón demanda de personal calificado y con conocimientos en esa área; la incorporación de material bibliográfico y guías prácticas durante el desarrollo académico de técnicos e ingenieros sobre esta temática, permitirán en un futuro un mayor aprovechamiento de esta tecnología en el ámbito local, al mismo tiempo mejorara las posibilidad de acceder a un mejor empleo
- SIP es un protocolo que se encuentra en constante desarrollo, se recomienda fomentar en los diferentes centros educativos, y lugares de trabajo la documentación de pruebas y aplicaciones realizadas con este protocolo, con el fin de ayudar a solventar las diferentes limitantes que este presenta.

Fuentes de Información

BIBLIOGRAFIA:

1. GONZALO CAMARILLO, 2002. **SIP DEMYSTIFIED** (1° ED). MCGRAW-HILL .CAMPUS DE LEGANÉS.
2. JONATHAN DAVIDSON, JAMES PETERS. 2000. **OVER IP FUDAMENTALES. CISCO PRESS**, MEXICO.INDIANA 46240
3. TAMARA RAMÍREZ ANDRADE, JAIME DÍAZ ROJAS .2006. **APLICACIONES SOBRE UNA RED DE TELEFONÍA IP**. UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA, ESPAÑA
4. BARRIE DEMSPETER, DAVID GOMILLON, SEPTEMBER 2005 **BUILDING TELEPHONY SYSTEMS WITH ASTERISK**, PACKT PUBLISHING.
5. JIM VAN MEGGELEN; JARED SMITH ,AGOSTO 2008, **ASTERISK™: THE FUTURE OF TELEPHONY, 2ND EDITION**, JARED SMITH; LEIF MADSEN PUBLISHER: O'REILLY

REFERENCIAS ELECTRÓNICAS

1. <http://www.digium.com/en/index.php>

Digium, Inc. All rights reserved (2008). VoIP Telephony Communication Solutions for Busines. [en línea]. Copyright © 2008 Digium, Recuperado Lunes 03 de marzo 2008.

2. <http://www.textoscientificos.com/redes/conmutacion>

Textos Científicos. (2007).Redes conmutadas. [en línea].Textos Científicos .España. Recuperados Lunes 20 de marzo 2008.

3. <http://www.voipforo.com/>

VoIP Foros (2008). SIP Vs. H.323 - Comparativa. [en línea]. SIP Vs. H.323 – Comparativa. Recuperado martes 29 de marzo 2008

4. <http://www.dednet.net/institucion/itba/cursos/000183/demo/unidad01/conmutaciondepaquetes.htm>

El planetamx Copyrigh (2000- 2001). Conmutación Por Paquetes. [En línea]. geocities.com/elplanetamx.mexico. Recuperado lunes 19 de Febrero 2008.

5. <http://www.ietf.org/rfc/rfc3261.txt>

Francisco Jiménez, Abril 04, 2006. Qué es VoIP. [En línea].Argentina, Recuperado martes 03 febrero 2008.

6. http://www.contactcentervoip.com/ES/ASTERISK/CENTRALITA/ASTERISK_P

Mayo 2007. Protocolos Asterisk SIP, MGCP, IAX. [En línea].Español, recuperada martes 19 marzo 2008.

7. <http://www.ietf.org/rfc/rfc3261.txt>

Rosenberg.2006. SIP: Session Initiation Protocol. [En línea].EEUU. Recuperación viernes 14 de marzo del 2008.

8. <http://medusa.unimet.edu.ve/procesos/referencias.html>

Landeau, R. (2004, 1 julio). *Referencias y citas bibliográficas* [material de apoyo]. Caracas, Venezuela: Universidad Metropolitana. Recuperado el 23 de octubre de 2006, de

9. <http://www.asterisk.org/support>

Pagina oficial de AsteriskNow, Copyright © 2008 Digium, Inc | Privacy Policy | Terms of Use Contact us | AsteriskNOW.org

Glosario

A.

AGI: (Interfaz de gateway asterisk): es una interfaz para agregar funcionalidades a Asterisk por medio de diferentes lenguajes de programación como ser Perl, PHP, C, Pascal, Bourne Shell, Java, etc, todo depende de lo que uno decida.

ATA: es un aparatito del tamaño de un modem de ADSL, que tiene dos conexiones: por una lado una conexión para un teléfono y del otro una conexión de red ethernet. No requieren de una PC para funcionar, son independientes, aunque obviamente necesitan de otro equipo de VoIP remoto para hablar :). Cumple la misma función que una PC con una placa FXS.

B.

BSD: (Distribución de Software Berkeley) se utiliza para identificar un sistema operativo derivado del sistema Unix nacido a partir de las aportaciones realizadas a ese sistema por la Universidad de California en Berkeley

C.

Códec: es una abreviatura de Codificador-Decodificador. Describe una especificación desarrollada en software, hardware o una combinación de ambos, capaz de transformar un archivo con un flujo de datos (stream) o una señal. Los códecs pueden codificar el flujo o la señal (a menudo para la transmisión, el almacenaje o el cifrado) y recuperarlo o descifrarlo del mismo modo para la reproducción o la manipulación en un formato más apropiado para estas operaciones.

Conexión E1: Es una Conexión de línea telefónica que puede transportar datos con una velocidad de hasta 1,920 Mbps. Según el estándar europeo (ITU), un E1 está formado por 30 canales de datos de 64 kbps más 2 canales de señalización. E1 es la versión europea de T1 (DS-1).

D.

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

DTMF: (Dual Tone Multifrequency) Multifrecuencia de doble tono. Tonos en diferentes hertz que utiliza una telefonía para marcar números. Cada número u opción del teléfono tiene su tono que es identificado en la telefonía.

E.

Enrutamiento: Se trata de la función de buscar un camino entre todos los posibles en una red de paquetes cuyas topologías poseen una gran conectividad.

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

F:

Firmware: es un bloque de instrucciones de programa para propósitos específicos, grabado en una memoria tipo ROM, que establece la lógica de más bajo nivel que controla los circuitos electrónicos de un dispositivo de cualquier tipo.

G.

Gatekeeper: Actúan en conjunto con varios Gateways, y se encarga de realizar tareas de autenticación de usuarios, control de ancho de banda, encaminamiento IP, etc. Es el núcleo de la red de telefonía IP.

Gateway: Es el elemento encargado de hacer de puente entre la red telefónica convencional (PSTN) y la red IP.

I.

IETF: (Grupo de Trabajo en Ingeniería de Internet) es una organización internacional abierta de normalización, que tiene como objetivos el contribuir a la ingeniería de Internet, actuando en diversas áreas, tales como transporte, encaminamiento, seguridad. Fue creada en EE.UU. en 1986.

ISUP es un protocolo de circuitos conmutados, usado para configurar, manejar y gestionar llamadas de voz y datos sobre PSTN.

IVR (Interactive Voice Response): basado en Asterisk permite la interacción con bases de datos internas del sistema (MySQL o PostgreSQL), externas como Oracle, DB2, Sybase, Informix, MS SQL Server o Web Services bajo SOAP.

M.

MIPS: es el acrónimo de "millones de instrucciones por segundo"

N.

Nodo: es un elemento activo de una red informática.

P.

Passthrough: es un conector que simplemente reenvía la señal, permite enviar una consulta SQL a una base de datos sin necesidad de pasar por el administrador de ODBC

PBX (Private Branch Exchange): Es una Central telefónica automática privada; que se conecta directamente a la red telefónica a través de conexiones E1 o T1.

PINT: servicio PINT define un protocolo para invocar ciertos servicios telefónicos de una red IP. Estos servicios incluyen realizar llamadas básicas, envío y recepción de faxes, y la recepción de contenidos en el teléfono. El protocolo se especifica como un set de mejoras y adiciones a los protocolos SIP 2,0 y SDP

Protocolo: es una descripción formal de un conjunto de reglas y convenciones que rigen la manera en que se comunican los dispositivos de una red.

PSTN (public switched telephone network): Red Publica de Conmutación Telefónica.

R.

Redes conmutadas: Es un grupo de dispositivos intermedios de comunicación, cuya función es trasladar los datos de un nodo a otro, sin importar el contenido que se esta trasmitiendo.

RTB (Red Telefónica Básica): Hace referencia al teléfono normal y es la típica conexión a INTERNET que conocemos.

RTP(Protocolo de Transporte de Tiempo real): Es un protocolo de nivel de transporte utilizado para la transmisión de información en tiempo real, como por ejemplo audio y vídeo en una video-conferencia.

S.

SDP: (Protocolo de Descripción de Sesión), es un formato para la descripción de medios de los parámetros de inicialización. Se ha publicado por la IETF como RFC 4566.

SIP Proxy: Actúa como un switch y redirector ultra rápido de llamadas con unas simples y pocas funciones en contraposición por ejemplo a Asterisk PBX donde se pueden switchear muchas menos llamadas por unidad de tiempo pero con cientos de funciones programables

Softphones: Son programas que permiten utilizar una pc con cualquier S.O. para llevar a cabo comunicaciones de voz o video sobre Internet.

Streaming: es un término que se refiere a ver u oír un archivo directamente en una página web sin necesidad de descargarlo antes al ordenador. Se podría describir como "hacer clic y obtener".

T

Transcoding: a la conversión directa (de digital a digital) de un códec a otro, en general con pérdida de calidad.

U

UAS: USER Agent Server, agente de usuario de servidor: responsable de aceptar las peticiones de establecimiento de sesiones recibidas

UAC: USER Agent client, agente de usuario de cliente: encargados de iniciar sesiones SIP.

URI: Uniform Resource Identifier: es un identificador más completo de las direcciones web, definido en RFC 2396.

URLs mailto: es un esquema que se usa para designar una dirección de correo de Internet de un individuo o servicio.

ANEXOS

Tabla de Abreviaturas

Abreviatura	Descripción
ATA	Adaptador Telefónico Analógico
DECT	Comunicación Digital Inalámbrica Mejorada
FXO	Foreing Exchange Office
FXS	Foreing Exchange Station
GSM	Sistema Global para Comunicaciones Móviles
IAX(2)	Protocolo de Intercambio de Asterisk (versión 2)
IETF	Grupo de Trabajo de Ingeniería de la Internet
ITU/UIT	Unión Internacional de Telecomunicaciones
IVR	Respuesta de Voz Interactiva Respuesta Vocal Interactiva
NAT	Traductor de Direcciones de Red
PBX (PABX)	Centralita Telefónica (Automática) Privada
PCM/MIC	Modulación por Impulsos Codificados
PSTN/RTB(C)	Red de Telefonía Básica (Conmutada)
QoS	Calidad de Servicio
RFC	Documento de Trabajo de Estandarización (Internet)
RTP	Protocolo de Tiempo Real
SCCP	Protocolo de Control de Llamadas Skinny
SIP	Protocolo de Señalización de Sesión(es)
SS7	Sistema de Señalización (versión) 7
TA/ATA	Adaptador Telefónico
UDP	User Data Protocol
VoIP	Voz sobre IP. Telefonía IP
VSAT	Terminal de Pequeña Apertura (Comunicaciones por Satélite)

- Software AsteriskNow en CD
- Softphone X-Lite en CD (en CD)
- Manual Usuario Polycom IP 301 SIP (en CD, Ingles)
- Manual adaptador análogo VoIP Solution MTA V 102(en CD, Ingles)

Aplicaciones Utilizadas Por Asterisk

AbsoluteTimeout:	Establece el tiempo máximo de una llamada (en segundos).	
Parametros:	SegundosValor en segundos de espera.
AddQueueMember:	Añade a una Cola de Llamadas la extensión desde la que se llama, o el agente o extensión determinada por el parámetro Interface. Además, es posible "penalizar" la extensión con un nivel de prioridad.	
Parametros:	NombreNombre de la cola de llamadas (Verificar que existe).
	Queue	
	InterfaceSe puede especificar un interface con el formato: Tecnología/recurso (o agent/identificador).
	PrioridadPenaliza la extensión con una prioridad.
ADSIProg:	Script con información que se pasa a la pantalla del terminal ADSI.	
Parametros:	ScriptNombre del script que se envía al terminal.
AgentCallbackLogin:	Login como Agente del sistema. La extensión recibe las llamadas del Agente especificado.	
Parametros:	NúmeroIdentifica al Agente que previamente debe estar dado de alta en "agents.conf".
	Agente	
	OpcionesOpciones de Agente.
AgentLogin:	Login como Agente del sistema, en modo exclusivo. Es decir, el teléfono siempre esta activo y escuchar un tono de beep cuando se le asigne una llamada.	
Parametros:	NúmeroIdentificador del Agente que debe de estar dado de alta en "agents.conf".
	Agente	
	OpcionesOpciones: s - no escucha el mensaje de bienvenida al sistema.
AgentMonitorOutgoing:	Muestra el Identificador del Agente que realiza una llamada usar con AgentCallbackLogin.	
Parametros:	OpcionesOpciones (n o d).
Agi:	Ejecuta un script o programa creado por el usuario.	

Parametros:	ScriptNombre Script.
	ArgumentosArgumentos que acompañan la llamada al script.
AlarmReceiver:	Recepción de Señal de Alarma desde un dispositivo especial.	
Parametros:	NINGUNO''
Answer:	Descuelga inmediatamente.	
Parametros:	NINGUNO''
AppendCDRUserField:	Añade el valor especificado en el campo User del registro de la llamada en el CDR.	
Parametros:	ValorValor que se inserta en el registro CDR.
Authenticate:	Pide al interlocutor que introduzca una clave válida para continuar.	
Parametros:	PasswordContraseña válida, que puede ser un archivo conteniendo contraseñas.
Background:	Reproduce una melodía o mensaje de audio mientras se espera la marcación de las teclas.	
Parametros:	ArchivoEspecifica el archivo que contiene la Melodía o Mensaje.
	OpcionesSkip no answer.
	LanguageSe puede seleccionar un language para seleccionar el idioma.
BackgroundDetect:	Reproduce una melodía o un mensaje mientras intenta detectar un dígito que corresponda con una extensión.	
Parametros:	ArchivoEspecifica el archivo que contiene la Melodía o Mensaje.
	PeriodoPeriodo de silencio en milisegundos (si sigue a un periodo de No Silencio salta a la extensión talk).
	Min. No-SilencioMínimo periodo de No-Silencio en milisegundos.
	Max. No-SilencioPeriodo de No-Silencio Máximo en milisegundos.
Blq_error_regla:	Reordena el índice de la extensión, insertando un salto de 101 posiciones sobre el índice que se señala en el parámetro "Índice Regla". A partir de esta posición las reglas son tabuladas hacia la derecha, y mientras no aparezca otra Aplicación BLQ_ERROR_REGLA, el índice sigue incrementándose. CUIDADO: con incluir índices iguales en la misma extensión. Se ejecutará la aplicación del primero de los	

índices repetidos.

Parametros: **Indice** **Regla**Señala la Regla que puede obligar a desviar el flujo de la llamada n+101 posiciones.
'0'

Busy: Devuelve la señal de ocupado hasta colgar o expirar el timeout.

Parametros: **Timeout**Tiempo que espera antes de colgar. En segundos.

Callingpres: Cambia la representación del CID en una llamada externa.

Parametros: **CID**Incluye el Número y el nombre del CID. "nombre" (Número).

capiNoES: Deja de aplicar la supresión de eco en un canal CAPI.

Parametros: **NINGUNO**''

ChangeMonitor: Cambia el Archivo donde se almacena el archivo de audio en un canal que se esté monitorizando.

Parametros: **Archivo**Nombre del Archivo dónde se almacenará ahora la grabación.

ChanIsAvail: Actualiza la vble. AVAILCHAN con el primer canal que esté libre. Se identifica un canal como: Tecnología/recurso, pudiendo anidar más canales añadiendo "& Tecnología/recurso ". Por ejemplo, SIP/100&SIP/101&SIP/102... para elegir el primero de los recursos SIP libres.

Parametros: **Lista Canales**Formato: Tipo_Canal/Recurso[&Tipo_Canal/Recurso].

CheckGroup: Controla que no se desborda el tope de las llamadas en el grupo activo (SetGroup).

Parametros: **Llamadas**Indica el máximo número de llamadas que admite el grupo actualmente.

Congestion: Reproduce el sonido de "Congestión en la línea" hasta que finalice la llamada o expire el valor de Timeout.

Parametros: **Timeout**Tiempo que espera antes de colgar. En segundos.

ControlPlayback: Reproduce un archivo, y se controla con algunas teclas su reproducción siempre que no coincida con una extensión.

Parametros: **Archivo**Nombre Fichero que contiene la melodía a reproducir.

SkipmsFinaliza la reproducción.

Tecla **de**Establece la tecla que permite un avance rápido de la audición.
avance

Tecla de rebobinar	deEstablece la tecla que permite retroceder en la audición.
Tecla de parada	Establece la tecla que permite parar la reproducción.
Tecla de pausa	Establece la tecla que permite parar la audición.

Cut: Permite asignar a una variable el valor de PARTE de otra. El resultado que se almacena es una subdivisión de la antigua variable, la división del parámetro "Bloque". La variable se divide en tantos trozos, como aparezca el delimitador.

Parametros:	Nueva VariableNombre de la Variable dónde se almacena el resultado.
	Vieja VariableLa Variable de dónde se desean extraer los datos.
	DelimitadorEl carácter separador a partir del cual se realiza el corte.
	BloqueEspecifica el bloque que se selecciona para asignar a la nueva variable.

DateTime: Reproduce una fecha a partir de una dada en formato unixtime. El formato unixtime representa una fecha expresada en la cantidad de segundos desde el 1 de enero de 1970 hasta el momento expresado en la fecha. La reproducción de las fechas se realiza de acuerdo a un formato determinado.

Parametros:	Fecha y HoraFecha y Hora expresada en formato "unixtime".
	Zona HorariaZona horaria (revisar zonas horarias en Linux).
	Formato ReproducciónFormato de reproducción de la fecha (por defecto: ABdY "digits/at" IMp).

DBdel: Elimina un dato de la base de datos interna, la cual se organiza en Familias de datos, y datos con valores dentro de la familia. De esta manera, para acceder a una variable, hay que especificar su Familia y el Valor determinado.

Parametros:	ValorValor a eliminar, indicándose en la forma (Familia/Valor).
--------------------	-------	--

DBdeltree: Elimina una Familia de datos de la base de datos interna. A veces es necesario indicar una clave para realizar esta operación.

Parametros:	FamiliaNombre de la Familia ([/Clave] si es necesario la password se añade de la forma indicada).
--------------------	---------	--

DBget: Obtiene el valor de un campo de la base de datos interna, especificado por la familia y el nombre del campo, y lo almacena en la variable especificada.

Parametros:	VariableVariable dónde se almacena el resultado de la consulta a la Base de datos interna.
	Campo BDNombre del Campo del que se extraé el dato, expresado con la forma "Familia/Campo".

DBput:	Inserta el valor especificado, dentro de un campo de la base de datos interna, especificado por la familia y el nombre del campo.	
Parametros:	Campo BDNombre del Campo que se inserta en la BD, expresado con la forma "Familia/Campo".
	ValorValor que contendrá el campo de la base de datos.
DeadAgi:	Ejecuta un script o programa creado por el usuario, al colgar la llamada en curso. (Observar también la situación de contexto Hangup).	
Parametros:	ScriptNombre Script.
	ArgumentosArgumentos que acompañan la llamada al script.
Dial:	Realiza una llamada a la extensión o número determinado en el campo "Num.Llamada", esperando una cantidad de segundos para que se establezca la comunicación. En caso de que no se establezca la comunicación pasará a la siguiente regla, pero si la comunicación no se establece por estar ocupado el interlocutor, saltará a la regla n+101.	
Parametros:	Canal ComunicaciónTipo de tecnología que se usa en la llamada: Zap, SIP, IAX, H323.
	Extension/NúmeroA una extensión interna sólo indicar la Extensión. Si es una llamada externa: Recurso/\${EXTEN} (o Número al que se llama).
	Segundos de esperaSegundos de espera a que se establezca la comunicación.
DigitTimeout:	Número de segundos válidos entre una y otra pulsación de los digitos.	
Parametros:	SegundosSegundos de espera.
Directory:	Accede a un servicio de directorio, apoyado en la información almacenada para la gestión de un contexto de Voicemail, en este caso el Apellido y Nombre de los usuarios. Con ello, se consigue que un usuario pueda acceder a la extensión "escribiendo el nombre" del usuario de la extensión a quien llama, de manera similar a como se dan de alta los nombres en una agenda de un móvil. SOLO FUNCIONA con aquellos usuarios que tengan activado el buzón de voz.	
Parametros:	Contexto VoicemailUsar: voip-h323.
	Contexto de LlamadaContexto de Llamada donde se sitúa la regla de llamada a la extensión.
	OpcionesOpciones: busca por el nombre. Por defecto busca por apellidos.
Disa:	Permite hacer una llamada al exterior y colgar; en caso de que descuelgue el llamante recibimos la llamada.	

Parametros:	ClaveClave de acceso (usar: "no-password" si no se desea).
	ContextoContexto dónde se trata la llamada al interior en caso de que descuelgue el interlocutor.
	Caller IDCaller ID. Formato "My Phone" (Número).
EAGI:		Ejecuta un script o programa creado por el usuario, con la particularidad de que EAGI permite controlar el flujo de audio.
Parametros:	ScriptNombre Script.
	ArgumentosArgumentos que acompañan la llamada al script.
Echo:		Permite escuchar el eco de lo que se está hablando.
Parametros:	NINGUNO''
EnumLookup:		Establece el valor de la variable ENUM de acuerdo a la extension pasada.
Parametros:	ExtensionExtensión que vá a ser modificada a su valor ENUM.
Eval:		Evalúa una variable y se asgina a otra nueva. Formato: Nueva Variable = Algo (que puede ser otra variable representada por \${Variable}).
Parametros:	VariableVariable que es actualizada.
	ValorValor con el que se actualiza, que puede ser otra variable (\${vble}).
Exec:		Ejecuta una aplicación de las de esta lista, cuando es invocada, de tal manera que no está establecida en el Dialplan de manera fija.
Parametros:	AplicacionCualquiera de las aplicaciones soportadas por Asterisk.
	ParametrosLos que soporte la aplicación seleccionada (deben separarse con comas, caundo exista más de uno).
Festival:		Reproduce una cadena de texto usando el software Festival.
Parametros:	TextoTexto a Reproducir.
	TeclaTecla que permite interrumpir la alocución y continuar el dialplan (usar "any" para cualquiera).
Flash:		Envía el evento Flash a la línea (es usado dentro de un AGI normalmente para producir una transferencia).
Parametros:	NINGUNO''
ForkCDR:		Inserta un nuevo Registro en la tabla CDR contando a partir de este instante el temporizador.

Parametros: NINGUNO''

GetCPEID: Obtiene de teléfonos ADSI la información CPE ID, para configurar al canal zaptel adecuadamente en operaciones "on-hook".

Parametros: NINGUNO''

GetGroupCount: Obtiene el número de llamadas al grupo de llamadas en un momento dado. El valor obtenido actualiza la variable GROUPCOUNT.

Parametros: Grupo LlamadasNombre del Grupo de Llamadas (@categoria, opcional)

Goto: Envía el flujo de la llamada a una determinada posición del dialplan, establecida por los parametros de Contexto + Extension + Prioridad. Entre los contextos hay que tener en cuenta que además de los creados por el usuario existen otros de creación automática: default controlado desde la pestaña "Control IVR"; voip-h323: controla las extensiones del sistema; outgoing que gestiona todas las extensiones de salida creadas dentro de la opción "Comunicación Exterior"; servicios contexto administrado desde la ventana homónima (Tener en cuenta las mayúsculas y minúsculas).

Parametros: ContextoContexto al que se desvía la llamada. Los contextos: default, servicios, outgoing, voip-h323; son creados por el sistema, y pueden referenciarse.
ExtensionEspecifica la extensión, dentro del Contexto, a la que se desvía (si no se especifica nada, irá a la extensión "s"-start.
PrioridadUltimo nivel de detalle al que se puede llegar, y expresa el índice de una regla dentro de las de la extensión.

GOTO-IF: Ejecuta la Sentencia Goto solo si se PULSA la opción especificada. En realidad, lo que realiza es la construcción de una extensión con una sola Regla: exten=>Opcion,1,Goto(Contexto,Extension,Prioridad). Sirve cuando no se quiere en realidad construir una extension con más reglas, como sucede en contextos de IVR, dónde se desea una regla para controlar la opción de salida.

Parametros: Opcion a TeclearSe especifica lo que debe.
ContextoContexto al que se redeirige el flujo de la llamada.
ExtensionExtensión de destino dentro del contexto señalado.
PrioridadPor último, el orden dentro de las reglas que gobiernan la extensión.

GotoIf: Permite un desvío del flujo de la llamada dentro de las reglas definidas para la misma extensión (ojo, no permite el salto directo a otro contexto o extensión). Además con la particularidad de que se vá a controlar el salto en función de la evaluación de una expresión condicional. Esta expresión condicional se conforma como: \${VARIABLE} [Elemento Condicional] Valor/\${Variable}. Teniendo en cuenta que hay que separar con un espacio en blanco los elementos.

Parametros:	<p>Condicion</p> <p>Regla de Salto True</p> <p>Regla de Salto False</p>	<p>....Expresión Condicional que será evaluada (separados por un espacio en blanco): \${VARIABLE} [Condicion] [\${Variable} o Valor].</p> <p>....Número de la Regla a la que salta en caso de ser Verdadera la Expresion.</p> <p>....Número de la Regla a la que salta en caso de ser Falsa la Expresion.</p>
GotoIfTime:	<p>Desvía el flujo de la llamada a un contexto y extensión especificados, SOLO cuando se esté en dentro del rango fecha-hora especificado en los parametros (Un símbolo * indica todo el rango posible en el parámetro).</p>	
Parametros:	<p>Rango Horario</p> <p>Dias Semana</p> <p>Dias del Mes</p> <p>Meses</p> <p>Regla de Salto True</p> <p>Regla de Salto False</p>	<p>....En formato de 24 Horas, y con una hora de comienzo y otra de final (Por ejemplo: 08-14. Las 24 horas se indican con *).</p> <p>....Indica un numero de Dias habiles en la semana (0 y 7 indican Domingo; * indica todos).</p> <p>....Indica que días del mes se desean (* indica todos).</p> <p>....Indica que meses se desean (* indica todos).</p> <p>....Número de la Regla a la que salta en caso de ser Verdadera la Expresion.</p> <p>....Número de la Regla a la que salta en caso de ser Falsa la Expresion.</p>
Hangup:	<p>Cuelga inmediatamente.</p>	
Parametros:	<p>NINGUNO</p>	<p>....''</p>
HasNewVoicemail:	<p>Esta aplicación permite determinar si el usuario del buzón tiene mensajes almacenados. Si no existen mensajes, el flujo de la llamada salta a la regla siguiente; mientras que el flujo de la llamada se desvía a la regla con el índice n+101 si existen mensajes en el buzón especificado.</p>	
Parametros:	<p>Número del Buzón</p> <p>Variable</p>	<p>....El Buzón se puede completar además con el contexto con "@contexto". Opcionalmente, se puede almacenar el número de mensajes existentes en una variable para ser consultada posteriormente.</p> <p>....El Número de Mensajes en el Buzón son almacenados en la variable especificada.</p>
ICES:	<p>Conecta a un servidor ice-cast de streaming. Necesita un archivo de configuración.</p>	
Parametros:	<p>Fichero</p>	<p>....Fichero de configuración.</p>

INCLUDE:	Direcciona a un contexto, de una forma especial, ya que al introducir una regla include, se toman en consideración los valores temporales, que se pueden modificar en los parámetros del contexto.	
Parametros:	ContextoContexto destino.
LookupBlacklist:	Mira si el CallID de la llamada está dentro de la lista negra de llamantes. Si está, salta a la regla n+101.	
Parametros:	NINGUNO''
LookupCIDName:	Mira si el CallID de la llamada está dentro de la lista "cidname". Si está, establece la variable CIDNAME al valor de la lista.	
Parametros:	NINGUNO''
Macro:	Ejecuta una Macro. Las Macros son Contextos especiales, en los cuales solo caben reglas para la extension "s" las cuales son ejecutadas en orden ... para posteriormente devolver el flujo de las acciones al siguiente paso en la llamada a esta aplicación Macro. Los argumentos dentro del contexto macro se referencian con $\${ARG1}$ $\${ARG2}$...	
Parametros:	Nombre ArgumentosNombre de la Macro a la que se llama como macro->nombre>.Los Argumentos se separan por comas en caso de existir más de uno.
MailboxExists:	Es una función que permite un salto condicional a la regla n+101. Comprueba si existe el Buzon de Voz. En caso de que exista salta a la regla n+101.	
Parametros:	Número BuzonNúmero del Buzon (seguido de "@contexto del buzón", opcional).
MeetMe:	Encamina una llamada a una sala de conferencia, dónde podrá escuchar la conversación de los que allí se agreguen, en función de la opción que se determine. Entre las principales opciones: a-Modo Administrador; s-Sólo escuchar; t-Sólo hablar; p-Salir pulsando #; M-escuchar música cuando esta sólo en la sala; x-cierra la sala cuando no haya nadie ... consultar más opciones en la documentación.	
Parametros:	Sala Opciones de EntradaIndicativo de la sala (un Número de Sala).Condiciones en las que se entra en la sala.
MeetMeAdmin:	Ejecuta un comando de administrador sobre la conferencia dada o sobre un usuario dado de la sala. Los comandos que se tienen: M-Activa el sonido; m- Pone en silencio (mute); L- Bloquea la sala; l- Desbloquea la sala; K- Sacar todos los usuarios de la sala; k- saca solo un usuario de la sala.	
Parametros:	Sala ComandoIndicativo de la sala (un Número de Sala).Uno de los indicados anteriormente (M m L l K k).

	UsuarioSi se especifica un usuario, se aplica el comando al usuario especificado (Solo comando k).
MeetMeCount:		Dice el número de participantes en la sala de conferencias especificada. Si además se especifica una variable, se almacena en la variable el valor hallado.
Parametros:	SalaIndicativo de la sala (un Número de Sala).
	VariableNombre de la Variable dónde se almacena el resultado.
Milliwatt:		Genera un tono constante de 1000Hz a 0dbm (mu-law).
Parametros:	NINGUNO''
Monitor:		Graba la comunicación en curso en un formato de Audio (wav, gsm, mp3) y la almacena en un archivo de audio la salida y en otro la entrada .. a menos que se seleccione la opción "m", en cuyo caso son mezclados los archivos en uno solo. Otra opción, "b" indica que no se quieren grabar los saltos de una extensión a otra, en caso de que se produzcan.
Parametros:	Formato AudioEspecifica si se desea almacenar el Audio en formato wav, gsm, mp3.
	ArchivoNombre con el que se almacena (se pueden usar nombre de Variables).
	Opcionesm- mezcla sonidos de entrada y salida; b-no graba mientras hay una transferencia a otro canal.
MP3Player:		Ejecuta a través del programa mpg123 un fichero o un stream de audio (especificar URL).
Parametros:	LocalizacionDirección URL
MusicOnHold:		Ejecuta la audición de la música en espera de la clase seleccionada. Las clases se definen dentro del archivo musiconhold.conf (Por defecto están las clases default y random. default es la que se activa si no se especifica ninguna clase). La clase default, puede tomar los valores de otra existente, mediante la aplicación SetMusicOnHold.
Parametros:	Clase MusicalValor: random o "no poner nada"
MYSQL:		Ejecuta una sentencia MySQL (Connect Query Fetch Clear Disconnect), incluso almacena el resultado en variables que pueden ser accesibles en el DialPlan. Observar su sintaxis: Connect Conector Lista_Vbles (dbhost dbuser dbpass dbname) Query Resultado Conector Consulta (SELECT algo FROM Tabla WHERE sentencia_sin_espacios) Fetch Resultado Conector Lista_Vbles (vble1 vble2 ...) Clear Conector Disconnect Conector.
Parametros:	SentenciaUna de las Siguietes: Connect Query Fetch Clear Disconnect.
	ResultadoVble en la que se almacena el resultado de la ejecución de la sentencia SQL. (Para Connect,Query,Fetch).

ConectorVariable que señala dónde buscar los datos a Query/Fetch/Clear/Disconnect. Siempre es `${variable_resultado}`.

Consulta SQL/Lista VConsulta SQL (Sin entre comillas, y separando el espacio en blanco como " ") o Lista de Variables donde almacenar (Fetch) o dónde se almacenan los datos de conexión (Connect).

NBScat: Reproduce un stream NBS local.

Parametros: **NINGUNO**''

NoCDR: No almacena la información de la llamada en la tabla CDR.

Parametros: **NINGUNO**''

NoOp: No ejecuta nada.

Parametros: **NINGUNO**''

Park: Auto Aparca al usuario en el espacio de "parking" de llamadas, normalmente para poder determinar como está ese espacio. Se usa junto a una transferencia supervisada. Debe existir el contexto "parkedcalls".

Parametros: **Extension**Número de la Extension correspondiente al Rango de "Parking" dónde se aparca la llamada.

ParkAndAnnounce: Aparca la llamada indicando dónde es aparcada, y que hacer si expira un tiempo máximo de segundos en espera sin ser atendida.

Parametros: **Avisar**Indicando que va a ser aparcado, con PARKED, o el Número de Extensión en la que se aparca.

TimeoutTiempo máximo en espera antes de ir al contexto de retorno.

Marcar Aviso de LlamPermite especificar una llamada de aviso a una extensión. Formato: Aplicación Dial. Ejemplo: SIP/\${EXTEN}.

Contexto de RetornoContexto al que retorna la llamada en caso de timeout. Formato: Contexto,Extension,Prioridad.

ParkedCall: [NULL].

Parametros: **Extension**Recupera una llamada aparcada previamente en la extensión especificada.

ParkedCall: [NULL].

Parametros: **Extension**Recupera una llamada aparcada previamente en la extensión especificada.

Playback:	[NULL].		
Parametros:	FicheroReproduce un archivo de audio, mientras no se cuelgue la llamada.	
Playtones:	[NULL].		
Parametros:	Tono registrado o LiReproduce un tono, dado desde una lista, o registrado en el archivo "indications.conf".	
PrivacyManager:	Pide el número de teléfono (10 cifras) en caso de no llevar CID.		
Parametros:	NINGUNO''	
Progress:	Indica que por este canal se puede conocer el progreso de la llamada ("in-band progress").		
Parametros:	NINGUNO''	
Queue:	Envia la llamada en curso a una cola de llamadas, previamente dada de alta. Además se puede acompañar de diversas acciones el desvío a la cola.		
Parametros:	Nombre de ColaCola de llamadas dada de alta en el sistema.	
	Opciones de LlamadatT-Permite Transferir la llamada; r-Escuchar Tono Ring (por defecto tT).	
	Direccion URLPermite ejecutar un comando via URL.	
	MensajeMensaje de anuncio de algun evento.	
	TimeoutTiempo maximo de espera a ser atendido en la Cola.	
Random:	Salto condicional basado en una probabilidad.		
Parametros:	ValorValor Probable (entre 1 y 100).	
	Contexto de SaltoFormato: contexto,extension,prioridad.	
Read:	Lee un numero de digitos (que finalizan con la tecla # o al llegar a un límite marcado) y se asignan a una variable.		
Parametros:	VariableNombre de la variable dónde se almacenará el resultado de la lectura.	
	ArchivoArchivo de audio con un mensaje, que se reproduce antes de la lectura.	
	Max. DigitosNúmero Máximo de dígitos permitidos en la lectura.	

	OpcionesSe aplican si la linea no está activa: skip- sigue en la siguiente regla; noanswer- lee datos aunque no esté la linea activa.
Record:		Graba en un archivo un mensaje de voz en el formato especificado.
Parametros:	Nombre FicheroNombre del archivo que almacena la grabación.
	FormatoFormatos de grabación: g723, g729, gsm, h263, ulaw, alaw, vox, wav, WAV.
	Max. SilencioSi detecta Silencio durante los segundos especificados, termina la grabación.
	Max. DuracionTiempo limite de la grabación, en segundos.
	OpcionesSe aplican si la linea no está activa: skip- sigue en la siguiente regla; noanswer- graba aunque no esté la linea activa.
RemoveQueueMember:		Nombre de la Cola de la que se desea dar de baja, como agente de ella. Si no se especifica nada, el sistema entiende que es desde la misma extensión desde dónde se llama, en caso de rellenar el campo Interface, se dará de baja lo especificado en ese parámetro.
Parametros:	ColaNombre de la Cola.
	InterfaceIndica el interface que quiere darse de baja. Ejemplo: SIP/\${CALLERIDNUM}.
ResetCDR:		Pone a Cero el registro con la información de la llamada en curso, que va a ser almacenado en la tabla CDR. La opción CDR almacena previamente el contenido.
Parametros:	OpcionCUIDADO!! Pone a cero el CDR. Opción: w, almacena todo.
ResponseTimeout:		Establece el tiempo de espera a la marcación de una extensión.
Parametros:	SegundosTiempo de espera.
Ringin:		Devuelve la señal de RING.
Parametros:	NINGUNO''
RxFAX:		Recibe un FAX y lo envía a un archivo. Modifica además las variables REMOTESTATIONID, FAXPAGES, FABITRATE, FAXRESOLUTION.
Parametros:	Nombre de ArchivoNombre del archivo que contiene lo que se vá a enviar vía fax.

SayAlpha:	Deletrea la cadena de texto. Esta cadena puede tener letras, números o los símbolos ! @ # \$ * - + = / (espacio en blanco).	
Parametros:	Cadena TextoCadena de Texto sin entrecomilla.
SayDigits:	Deletrea los dígitos pasados, en el language activo en ese canal.	
Parametros:	NúmeroNúmero a Deletrear ... puede ser el contenido de una variable.
SayNumber:	Dice el Número pasado, en el language activo en ese canal.	
Parametros:	NúmeroNúmero que se leerá (entre 0 y 99.999.999). Sin decimales.
	GéneroEntonación del audio: m-masculino; f-femenino; c-entonación de ordenador.
SayPhonetic:	Deletrea la cadena pasada, usando el alfabeto fonético que es usado por la NATO.	
Parametros:	Cadena TextoCadena de texto a deletrear.
SayUnixTime:	Nos dice la fecha y hora en función del valor pasado en formato unixtime (segundos desde 1/ene/1970).	
Parametros:	UnixTimeValor temporal en formato unixtime.
	Zona HorariaZona Horaria como las especificadas en Linux.
	Formato de dictadoForma en la que se escuchará la fecha dada. Ejemplo: ABdY "digits/at" IMp.
SendDTMF:	Envía una cadena de dígitos, a través de la señal DTMF establecida en el canal.	
Parametros:	DígitosDígitos (válidos: 0-9 *#abcd).
SendImage:	Envía una cadena de dígitos, a través de la señal DTMF establecida en el canal.	
Parametros:	ImagenArchivo con imagen.
SendText:	Envía un mensaje de texto al cliente. Si el cliente no soporta la recepción de texto, el dialplan continuará en la regla n+101, si existe.	
Parametros:	MensajeMensaje de texto.
SendURL:	Envía una dirección URL al cliente, siempre que lo soporte. En caso de no soportarlo, se desvía la ejecución a la regla n+101.	
Parametros:	URLDirección URL.
	Opcioneswait - indica que espera a que se haya cargado la URL antes de continuar.

SetAccount:	Establece el valor del campo Account en el CDR, para control de llamada.	
Parametros:	ValorValor que se introduce en el campo Account (puede ser el contenido de una variable).
SetAMAFlags:	Establece el valor del campo amaflags en el CDR, para control de llamada.	
Parametros:	FlagsValores posibles: default, billing, documentation, omit.
SetCallerID:	Establece un nuevo valor para el CallerID de la llamada.	
Parametros:	CIDNuevoCID.
SetCallerPres:	Presentacion del CallerID de la llamada: allowed_not_screened allowed_passed_screen allowed_failed_screen allowed prohib_not_screened prohib_passed_screen prohib_failed_screen prohib unavailable.	
Parametros:	PresentacionValores: allowed_not_screened allowed_passed_screen allowed_failed_screenallowed prohib_not_screened prohib_passed_screen prohib_failed_screen prohib unavailable.
SetCDRUserField:	Establece el valor del campo User en el registro CDR para la llamada en curso.	
Parametros:	ValorValor que se inserta en el campo User.
SetCIDName:	Establece un nuevo valor para la parte del Identificador de Nombre (no el número) del CallerID de la llamada.	
Parametros:	NombreNuevo Nombre asignado a la parte del nombre dentro del CID.
SetCIDNum:	Establece un nuevo valor para la parte del Número (no el nombre) del CallerID de la llamada.	
Parametros:	Número de TeléfonoNuevo Número telefónico.
SetGlobalVar:	Establece una variable Global, que se puede usar en otro canal de comunicación al ser de ámbito global a todo el Dialplan.	
Parametros:	Variable GlobalNombre de la Variable Global.
	ValorValor al que se establece.
SetGroup:	Especifica la creación de un grupo para control de acceso. Por ejemplo, muy útil para limitar el numero de llamadas a una extensión.	
Parametros:	GrupoNombre del grupo (puede ser una extensión o una variable).

SetLanguage:	Establece el directorio por defecto de donde coger los archivos de voz para los mensajes. Por defecto es ingles (en), para el idioma español (es) .. se delimitan por abreviaturas del pais, por lo general.	
Parametros:	PaisAbreviatura del language, según pais.
SetMusicOnHold:	Establece la "clase" musical por defecto, para MusicOnHold.	
Parametros:	ClaseClase Musical definida en el archivo musiconhold.conf.
SetVar:	Establece el valor de una variable global, que se puede usar durante la comunicación de cualquier canal.	
Parametros:	VariableNombre de la variable global.
SetVar:	[NULL].	
Parametros:	ValorValor al que se establece.
SIPDtmfMode:	Establece el parametro dtmfmode para una llamada a través de un canal SIP. Valores válidos: inband,info, rfc2833.	
Parametros:	DTMF ModeNuevo valor de señalización de la señal dtmf, en el canal de comunicación en curso.
SoftHangup:	Cuelga un canal en curso, aunque hay que saber con anterioridad el identificador correcto de la comunicación, que suele ser: Tecnología/Canal-recurso (por ejemplo: SIP/701-af23). Formato: Tipo_Canal/identificador_recurso.	
Parametros:	CanalFormato: Tipo_Canal/identificador_recurso.
StopMonitor:	Finaliza la grabación de la llamada en curso.	
Parametros:	NINGUNO''
StopPlaytones:	Finaliza la reproducción de la lista de tonos especificada en Playtones.	
Parametros:	NINGUNO''
StripLSD:	Trunca por la Izquierda el número o máscara de la extensión actual.	
Parametros:	DigitosNúmero de Dígitos que seran eliminados.
StripMSD:	Trunca por la Derecha el número o máscara de la extensión actual.	

Parametros:	DigitosNúmero de Dígitos que seran eliminados.
Suffix:	Añade los digitos especificados al final del número o máscara actual.	
Parametros:	NúmerosNúmero que será añadido al valor $\${EXTEN}$, para formar el número: $\${EXTEN}$ "Números".
System:	Ejecuta una aplicación en Linux.	
Parametros:	AplicaciónAplicacion Linux como si se lanzase desde el intérprete de comandos.
TestClient:	Ejecuta un test sobre el cliente dado, almacenando el resultado en "/var/log/asterisk/testreports/testid.txt".	
Parametros:	Cliente''
TestServer:	Ejecuta un test sobre Servidor y almacena el resultado en "/var/log/asterisk/testreports/testid.txt".	
Parametros:	NINGUNO''
Transfer:	Transfiere la llamada en curso a otra extensión, siempre que esté dentro de su mismo Contexto. En caso de ser fallida, salta a la regla n+101.	
Parametros:	ExtensionNúmero de extensión a la que se pretende realizar el Salto.
TrySystem:	Ejecuta una aplicación en Linux. Si existe un error en la ejecución sigue en la regla n+101.	
Parametros:	AplicaciónAplicacion Linux como si se lanzase desde el intérprete de comandos.
TxFAX:	Envia el Archivo especificado. Ademas usa las variables LOCALSTATIONID y REMOTESTATIONID.	
Parametros:	ArchivoNombre del Archivo en formato TIFF.
TXTLookup:	Busca el interlocutor que llama a partir del campo de texto via DNS. en la base de Datos Interna. Actualiza la variable TXTCIDName.	
Parametros:	CIDIdentificador de la llamada (puede ser una variable).
UserEvent:	Envía al subsistema "Manager" un evento que podrá ser usado por el usuario. Como parámetros se pasa al Manager, un nombre de evento y opcionalmente un campo con más opciones.	
Parametros:	NombreNombre de Evento.
	Cuerpo deLas Opciones posibles que se pueden gestionar desde el manager.

Evento

Verbose: Nivel de detalle de la presentación de mensajes en consola.

Parametros:

NivelDe 0 hasta 7.
MensajeMensaje Personalizado que será presentado en los archivos de "log".

VoiceMail: Desvía al buzón de voz de la extensión especificada, la llamada en curso. Es posible especificar el motivo del desvío, mostrando un mensaje según se indique en el campo de Mensaje de Aviso(b=ocupado, u=no disponible), o no emitir ninguna alocución previa a la grabación del mensaje (s=Sin Mensaje previo).

Parametros:

Mensaje de Avisob- ocupado; u- no disponible; s- sin mensaje.
ExtensiónNúmero de Extension (y en algunos casos la definición del contexto al que pertenece mediante "@contexto", opcionalmente).

VoiceMail2: Desvía al buzón de voz de la extensión especificada, la llamada en curso. Es posible especificar el motivo del desvío, mostrando un mensaje según se indique en el campo de Mensaje de Aviso(b=ocupado, u=no disponible), o no emitir ninguna alocución previa a la grabación del mensaje (s=Sin Mensaje previo).

Parametros:

Mensaje de Avisob- ocupado; u- no disponible; s- sin mensaje.
ExtensiónNúmero de Extension (y en algunos casos la definición del contexto al que pertenece mediante "@contexto", opcionalmente).

VoiceMailMain: Accede al sistema de control del buzón de voz. El subsistema de Buzón de voz, le indicará mediante un menú de voz como acceder a todas las funciones. Se puede acceder directamente a un buzón específico, señalando el campo "Buzon de Voz", con el valor de la extensión a la que acceder. Si no, si se deja en blanco, accede al menú principal de la aplicación, pidiendo los datos necesarios. Si se indica "s", en el campo Mostrar Mensaje, no será reproducido el mensaje de bienvenida.

Parametros:

Mostrar mensajesólo cabe dejarlo en blanco, o poner el valor "s".
Buzon de VozExtensión(@contexto) del buzón de voz a analizar. Si no se indica nada, accede al menú principal.

VoiceMailMain2: Accede al sistema de control del buzón de voz. El subsistema de Buzón de voz, le indicará mediante un menú de voz como acceder a todas las funciones. Se puede acceder directamente a un buzón específico, señalando el campo "Buzon de Voz", con el valor de la extensión a la que acceder. Si no, si se deja en blanco, accede al menú principal de la aplicación, pidiendo los datos necesarios. Si se

indica "s", en el campo Mostrar Mensaje, no será reproducido el mensaje de bienvenida.

Parametros:	Mostrar mensajesólo cabe dejarlo en blanco, o poner el valor "s".
	Buzon de VozExtensión(@contexto) del buzón de voz a analizar. Si no se indica nada, accede al menú principal.
Wait:		Espera un tiempo expresado en segundos antes de continuar.
Parametros:	SegundosSegundos de Espera.
WaitExten:		Espera los segundos expresados, que se marque una nueva extensión antes de continuar.
Parametros:	TiempoSegundos de Espera.
WaitForRing:		Espera los segundos especificados a recibir la señal de Ring.
Parametros:	TimeoutValor en segundos.
WaitMusicOnHold:		Durante el tiempo que se especifique se retiene la llamada con Musica, la que esté establecida en la clase "default".
Parametros:	TiempoSegundos de espera.
Zapateller:		Genera un tono que bloquea a los televidentes que llamen. Opciones: answer-responde despues del tono, nocallerid-genera el tono si no hay CID.
Parametros:	Opcionesvalores posibles: Answer o nocallerid.
ZapBarge:		Controla un canal Zap, que esté siendo monitorizado.
Parametros:	CanalCanal zap indicado por Zap/recurso.
ZapRAS:		Permite acceso RAS a otro servidor usando una conexión pppd -en una linea RDSI-. Solo funciona para lineas RDSI controlado por un driver Zaptel.
Parametros:	ArgumentosLa lista de argumentos debe separarse con el símbolo " ".
ZapScan:		Permite el control de los canales Zap, para encaminarlos adecuadamente. (#-selecciona siguiente canal; *-termina.
Parametros:	NINGUNO''