
**UNIVERSIDAD AUTÓNOMA DEL ESTADO
DE HIDALGO**

**INSTITUTO DE CIENCIAS BÁSICAS E INGENIERÍAS
INGENIERÍA EN ELECTRÓNICA Y TELECOMUNICACIONES**

**IMPLEMENTACION DE UN SERVIDOR DE
COMUNICACIONES BASADO EN SIP**

T E S I S
QUE PARA OBTENER EL TÍTULO DE
INGENIERO EN ELECTRÓNICA Y TELECOMUNICACIONES

P R E S E N T A N

TELESFORO MORALES GARNICA

ALFREDO LUGO GARCIA

ASESOR DE TESIS

MTRO. MIGUEL ÁNGEL HERNÁNDEZ VÁZQUEZ

PACHUCA DE SOTO, HIDALGO. JUNIO DE 2007

Dedicatoria

A mis padres, que siempre me han brindado su apoyo incondicional en las buenas y en las malas, que siempre y a pesar de todo han confiado totalmente en mí y que por sobre toda adversidad me han forjado una vida a plenitud y llena de oportunidades pues me han inculcado el valor más grande de la vida, el respeto. Mi admiración, mi respeto y todo el amor de mi corazón para ustedes Don Tele y Doña Chayo.

A mis hermanas, quienes han sabido apoyarme sobremanera, quienes han sido los soportes de mis proyectos y el empuje de mis logros, quienes me han enseñado que valemos más por lo que somos que por lo que tenemos, a ustedes hermanas por darme la felicidad de tener conmigo a Joqsani, Diego, Angie y Samuel. Con todo mi amor y cariño a ustedes Betty, Vilma y Rosario.

A la persona que siempre me ha apoyado incondicionalmente, quien me ha enseñado el poder que tiene una sonrisa y lo valioso que es contar con alguien para quien eres el motivo de triunfar, a la persona que me ha permitido entrar a su corazón, con todo mi cariño y mi corazón a ti Ana Laura.

Telésforo

Dedicatoria

A mis Padres a quienes amo infinitamente, que por siempre han sido la fuente más importante de motivación en mi vida, por ser la guía y el soporte en cada uno mis logros incluyendo esta tesis, ya que sin sus sabios consejos, valores, alientos y regaños muchos o ninguno de estos los hubiera conseguido, estoy seguro de que en ningún lado encontraré la paciencia que ustedes me han tenido y mucho menos la confianza que todo la vida me han brindado, por todo esto, por estar siempre a mi lado y por ser los encargados de traerme al mundo GRACIAS.

A mi hermana Diana y a mi sobrino Fernando por el gran amor que les tengo, quiero agradecerles por motivarme a ser el mejor ejemplo para ustedes, por quererme y estar siempre a mi lado, de igual manera quiero que tengan en cuenta que jamás los dejaré solos y siempre contarán conmigo.

A quien ha compartido mucho de su valioso tiempo conmigo, quien ha puesto toda su confianza en mí, quien me ha apoyado en este gran logro tan importante para mí y quien me ha hecho un espacio en su corazón, todo esto de manera incondicional, muy especialmente y con todo mi corazón, mil gracias Claudia.

Alfredo

Prefacio

La telefonía IP es una tecnología que promete revolucionar la forma en como nos comunicamos, a la par que maneja un decremento en los costos de esta comunicación e incrementa la velocidad, rentabilidad y disponibilidad de su servicio. VoIP es una de las vertientes de la telefonía IP que ofrece la mayor capacidad de interacción en las redes IP que se utilizan en la actualidad.

Por lo tanto, en este proyecto se diseña e implementa un servidor de comunicaciones que utiliza la tecnología VoIP, permitiendo de esta manera, dar solución a los problemas de infraestructura, costos, disponibilidad y operatividad de la telefonía convencional.

Por ello, es importante que se conozca la evolución que ha tenido la telefonía desde el surgimiento de la PSTN, los inicios de comunicaciones internas a través de intercomunicadores hasta llegar a los primeros PBXs que daban solución al uso de la red pública para la comunicación interna.

A través de este proyecto se encontrará también la descripción detallada de lo que es la telefonía IP y VoIP, incluyendo las características operativas y requerimientos técnicos de estas tecnologías, lo que nos permitirá comparar la telefonía tradicional con la nueva generación de telefonía IP y VoIP.

De igual manera se podrán conocer las particularidades del protocolo SIP, utilizado en la implementación del servidor de comunicaciones, esto es, su estructura, la interacción que tiene con el estándar del modelo OSI, la forma en como se presentan los mensajes de comunicación y las respuestas para cada solicitud de comunicación.

El punto de mayor importancia dentro de este proyecto es dar a conocer la forma en como se implementó el servidor de comunicaciones basado en SIP, la utilización de recursos sin costo como lo es el software libre para el S.O. y para los softphones, además del PBX en software que se utilizó para diseñar un red completa de telefonía IP con su sistema de marcado y respaldos de comunicación, además de los llamados servicios digitales como el correo de voz y la llamada en espera.

Índice general

1. Metodología	1
1.1. Antecedentes	1
1.2. Problemática	1
1.3. Solución propuesta	2
1.4. Objetivos	3
1.4.1. Objetivo general	3
1.4.2. Objetivos específicos	3
1.5. Organización	3
2. PBX	5
2.1. Introducción	5
2.2. ¿Como surge la idea de un PBX?	5
2.2.1. Telefonía privada	5
2.2.2. Intercomunicador en paralelo	6
2.2.3. Intercomunicador con señalización selectiva	7
2.2.4. Circuito Jefe-Secretaria	8
2.2.5. Primeros PBX (Private Branch eXchange)	9

2.3. ¿Que es un PBX?	9
2.3.1. CPU (Central Process Unity)	11
2.3.2. PBX más modernas	13
2.3.3. Capacidad de la PBX	13
2.3.4. Funciones especiales	13
2.3.5. Recomendaciones para la administración de un PBX	14
 3. Telefonía IP y VoIP	 17
3.1. Introducción	17
3.2. La telefonía IP en la actualidad	17
3.3. Sistema tradicional de telefonía	18
3.3.1. Procesamiento de llamadas	18
3.3.2. Conexión entre centrales	19
3.3.3. Ruteo, señalización y protocolos	20
3.4. Telefonía IP	21
3.4.1. Codificación de la voz	22
3.4.2. Señalización	22
3.4.3. ¿Como funciona la telefonía IP?	23
3.5. La telefonía IP y las telecomunicaciones	24
3.6. Tecnología VoIP	25
3.6.1. Elementos de VoIP	27
3.6.2. Características de VoIP	28
3.6.3. Protocolos de VoIP	28

3.6.4. Aplicaciones de VoIP	29
3.7. Comparación de VoIP y la telefonía tradicional	32
3.7.1. Ventajas	32
4. Protocolo de inicio de sesión (SIP)	35
4.1. Introducción	35
4.1.1. Antecedentes de SIP	35
4.2. ¿Qué es SIP?	37
4.3. Línea de protocolos de Internet	38
4.3.1. Capa física	38
4.3.2. Capa de Internet	38
4.3.3. Capa de transporte	40
4.3.4. Capa de aplicación	43
4.4. Métodos de solicitud	44
4.5. Tipos de respuesta	62
4.6. Comparación con otras tecnologías	81
4.6.1. Introducción a H.323	81
4.6.2. Comparación de H.323 con SIP	88
5. Implementación del servidor SIP	91
5.1. Introducción	91
5.2. Linux como plataforma para SIP	91
5.2.1. Distribuciones de Linux	92

5.2.2. Distribución SUSE	93
5.3. PBX Asterisk	94
5.3.1. Hardware Soportado	94
5.3.2. Características	94
5.3.3. Protocolos soportados	94
5.3.4. Arquitectura de Asterisk	95
5.4. Instalación de Linux	96
5.5. Instalación de Asterisk	97
5.5.1. Código fuente de Asterisk	98
5.5.2. Extracción del código fuente	99
5.5.3. Compilación de Zaptel	99
5.5.4. Compilación de Libpri	100
5.5.5. Compilación de Asterisk	100
5.5.6. Compilación de Asterisk-sounds	101
5.6. Comandos CLI	101
5.7. Configuración de SIP a nivel local	103
5.7.1. Determinación de características para conexión a nivel local . .	104
5.7.2. Configuración de Asterisk SIP.conf	106
5.7.3. Configuración de Asterisk extension.conf	109
5.7.4. Configuración de Asterisk voicemail.conf	112
5.7.5. Configuración de los clientes	113
5.7.6. Conectividad entre clientes Windows, Linux y Servidor SIP . . .	115

5.8. Utilidad y ventajas del protocolo SIP	128
5.8.1. Convergencia e interoperabilidad del protocolo SIP	128
5.8.2. Nivel de utilidad del protocolo SIP	130
6. Conclusiones	133
A. Acrónimos	137
Bibliografía	145

Índice de Tablas

4.1. Ejemplo de solicitud INVITE	46
4.2. Cabeceras de campo obligatorio en una solicitud INVITE	47
4.3. Ejemplo de solicitud de registro para el usuario EUCLID	48
4.4. Cabeceras de campo obligatorias en una solicitud REGISTER	48
4.5. Ejemplo de solicitud BYE	49
4.6. Cabeceras de campo obligatorio de una solicitud BYE	49
4.7. Cabeceras de campo obligatorio en una solicitud ACK	50
4.8. Cabeceras de campo obligatorio en una solicitud CANCEL	50
4.9. Ejemplo de un mensaje de solicitud REFER	52
4.10. Cabeceras de campo obligatorio para una solicitud REFER	53
4.11. Ejemplo de una solicitud de SUSCRIBE	54
4.12. Paquetes de eventos y paquetes de plantillas	54
4.13. Cabeceras de campo obligatorio para una solicitud SUSCRIBE	54
4.14. Ejemplo de una solicitud NOTIFY	55
4.15. Cabeceras de campo obligatorio en una solicitud NOTIFY	56
4.16. Ejemplo de una solicitud de MESSAGE	57

4.17. Cabeceras de campo obligatorio en una solicitud MESSAGE	58
4.18. Ejemplo de solicitud INFO	58
4.19. Cabeceras de campo obligatorio en una solicitud INFO	59
4.20. Ejemplo de intercambio de respuestas provisionales	60
4.21. Cabeceras de campo obligatorio en una solicitud PRACK	61
4.22. Ejemplo de un mensaje UPDATE	61
4.23. Cabeceras de campo obligatorio en una solicitud UPDATE	62
4.24. Clases de respuesta SIP	63
4.25. Ejemplo de la respuesta 401	69
4.26. Ejemplo de respuesta 407	71
4.27. Ejemplo de la respuesta 483	75
4.28. Familia de estándares ITU 32x	82
4.29. Protocolos referidos por H.323	84
5.1. Información general de los clientes locales	105
5.2. Parámetros establecidos en la definición para SIP.CONF	108
5.3. Parámetros configurados en cada uno de los clientes SIP	115

Índice de figuras

2.1. Llamada por central pública	6
2.2. Intercomunicador en paralelo	7
2.3. Señalización selectiva	8
2.4. Circuito jefe-secretaria	8
2.5. Teléfonos secretariales	9
2.6. PBX private branch exchange	10
3.1. Sistema tradicional	21
3.2. Sistema de telefonía IP	24
3.3. Tecnología VoIP	26
4.1. Linea de protocolos multimedia de Internet	38
4.2. Abriendo y cerrando una sesion SIP	41
4.3. Abriendo conexiones TLS	42
4.4. INVITE con cuerpo de mensaje	45
4.5. Flujo de llamada para una REFER(referencia)	52
4.6. Llamada de un SUSCRIBE y NOTIFY	53
4.7. Mensajeria instantanea SIP	57

4.8. Uso de respuestas provisionales confiables	59
4.9. Llamada QUEVING	65
4.10. Respuesta 183 SESSION PROGRESS	66
4.11. Resolucion de re-INVITE simultaneas	78
4.12. Elementos pricipales de una red H.323	83
4.13. Flujo de llamade H.323	85
4.14. Secuencia de terminacion de llamada H.323	87
5.1. Especificaciones técnicas de los clientes	106
5.2. Interconexion de clientes SIP Y PBX asterisk en una red local	107
5.3. Pantalla de configuracion softphone X-LITE	114
5.4. Registro de cliente en un servidor sip	116
5.5. Solicitud de llamada de ext 110 a ext 115	116
5.6. Llamada entre máquina 1 y máquina 2 establecida	117
5.7. Terminacion de llamada entre máquina 1 y máquina 2	117
5.8. Solicitud de llamada máquina 2 a máquina 1	119
5.9. Llamada entre máquina 2 y máquina 1 establecida	120
5.10. Llamada entre máquina 2 y máquina 1 terminada	120
5.11. Softphone mostrando que hay nuevo correo de voz	125
5.12. Clientes estableciendo llamada con el correo de voz	125

Capítulo 1

Metodología

1.1. Antecedentes

Anteriormente era muy difícil pensar o imaginar, que las aportaciones de grandes investigadores como Edison o Bell, que contribuyeron al logro de la telefonía tradicional, se convertirían al paso de los años en el sistema universal de comunicación entre la humanidad, gracias a una gigantesca red de circuitos que cubren la tierra y que hacen posible el establecimiento de la conexión entre dos personas.

Hoy en día, todas las empresas tienen la necesidad de contar con servicios de telefonía, puesto que es muy complicado que tengan un buen funcionamiento si no lo hacen. La telefonía en general debe ser considerada en el mundo de la comunicación como uno de los implementos más importantes de la historia.

Lo anterior, no significa que haya terminado la evolución de la telefonía, si no todo lo contrario, parece ser el comienzo de una nueva era en donde, nacen nuevas tecnologías, que con el pasar del tiempo, seguirán facilitando la comunicación mundialmente.

1.2. Problemática

La problemática que presenta la telefonía tradicional es:

- El costo por infraestructura, se refiere a tener una red dedicada únicamente a datos y otra red exclusiva para voz, lo que duplica el costo en cableado y dispositivos para mantener la infraestructura.
- Otra problemática es el costo que representa hacer uso del servicio mediante alguna compañía proveedora, debido a las tarifas que manejan las compañías dependiendo del lugar a donde se desea la comunicación y el tiempo que dure la misma.
- Un inconveniente más, es la administración de dos redes diferentes, ya que implica más equipo y personal especializado en cada área, ya sea de voz ó de datos, además de aumentar el tiempo de respuesta ante un incontingencia.
- El utilizar equipos que trabajan mediante conmutación de circuitos implica un mayor consumo de energía, que se ve reflejado en el gasto de electricidad.
- Una limitación más es la falta de flexibilidad para movilizar los equipos telefónicos, ya que cada punto de la red telefónica equivale a una configuración en particular que hay que asignar a los dispositivos.

1.3. Solución propuesta

Diseñar e implementar un servidor de comunicaciones utilizando el protocolo SIP (Session Initiation Protocol), que mediante el uso de la red de datos ya establecida en alguna empresa o en cada hogar, permita realizar llamadas telefónicas entre usuarios, eliminando con esta implementación las limitantes de movilidad, costo y de cableado que presentan la red de telefonía actual.

La implementación que se desarrolla utiliza software libre, sin costo alguno, para el servidor de comunicaciones, de igual forma utiliza software gratuito para los teléfonos o softphones. La interoperabilidad es una característica fundamental de este trabajo, así que el servidor opera sobre diferente plataformas de sistema operativo, de manera que no tiene ningún problema de compatibilidad o de portabilidad

1.4. Objetivos

1.4.1. Objetivo general

Implementar un servidor de comunicaciones que proporcione el servicio de comunicación telefónica, entre usuarios finales en cualquier red IP, ofreciendo las ventajas de flexibilidad e interoperabilidad que caracterizan al SIP.

1.4.2. Objetivos específicos

Mostrar las ventajas de la telefonía IP sobre la PSTN(red conmutada de telefonía pública).

Demostrar que es más factible administrar una sola red para voz y datos a dos redes diferentes una para datos y otra para voz de igual forma mostrar el ahorro que representa el tener menos dispositivos físicos.

Mostrar la flexibilidad de la comunicación VoIP utilizando el protocolo SIP que permite enlazarse desde cualquier lugar a una misma extensión o número telefónico.

Dar un panorama amplio y general del protocolo SIP(protocolo de inicio de sesión) utilizado en el servidor de comunicaciones desarrollado en este trabajo.

Resumir cuales son las ventajas y alcances del SIP sobre otros protocolos utilizados en VoIP(voz sobre IP).

1.5. Organización

Por las características determinantes de voz sobre el protocolo de Internet en esta tesis se elige esta tecnología por el buen desempeño que ha presentado, sabiendo que la calidad en los servicios cumplirá con las expectativas en demanda de comunicación, ya que las facilidades que presenta la telefonía tradicional serán superadas. Para comprender a profundidad esta tesis se ha dividido en 5 capítulos cubriendo la siguiente temática:

En el capítulo 1 se hace una introducción general a los alcances y expectativas de esta tesis. Para el capítulo 2 se habrá de detallar la funcionalidad y características de un PBX (Private Branch eXchange) convencional utilizado en la PSTN (Public System Telephony Network), además de explicar la evolución de la telefonía tradicional pasando por los intercomunicadores hasta llegar a los PBXs en software.

Para continuar en el capítulo 3 se describen los antecedentes de la telefonía IP, además que se explicará la tecnología de VoIP (Voice over IP), su funcionamiento, infraestructura características, además de una comparación con otras tecnologías.

Posteriormente en el capítulo 4 se presentan las generalidades del protocolo SIP, utilizado en esta tesis, además de su funcionalidad en cuanto telefonía, servicios multimedia y mensajería instantánea, la manera en la que se estructura un mensaje de llamada en SIP, incluyendo los tipos de respuesta para cada solicitud y comparación con otros protocolos.

Finalmente en el capítulo 5 se explica el proceso que se siguió en la implementación del servidor desarrollado, se hace referencia a la descarga e instalación de la plataforma para el sistema operativo, instalación del PBX en software, diseño y configuración de canales SIP y plan de marcado, terminando con la configuración de clientes y pruebas de operatividad.

Capítulo 2

PBX

2.1. Introducción

En el presente capítulo, se mencionan de forma resumida los inicios de la telefonía privada para así visualizar claramente las generalidades de un PBX (Private Branch exchange), de esta forma se entenderá la funcionalidad de la implementación que se desarrolla.

2.2. ¿Como surge la idea de un PBX?

Surge con la intención de comunicarse de manera privada dentro de una comunidad cualquiera. Por lo tanto, para entender el concepto de comunicación privada, se debe de iniciar comentando como es que nacen las líneas telefónicas privadas [1].

2.2.1. Telefonía privada

La gran necesidad de comunicarse en el mismo edificio, comunidad, oficina o fábrica, dio paso al nacimiento de la telefonía privada. En los inicios cuando alguna persona quería establecer una comunicación, aún cuando fuera al cuarto contiguo a su ubicación, tenía que llamar por medio de la central pública [1].

Por ejemplo: en la Figura 2.1 se muestra como una secretaria para hacer una consulta con almacén tomaría la línea y marcaría el número del almacén a través de la central pública, con lo cual se genera un cargo que al acumularse hacen el costo de comunicación muy caro y poco rentable, imagínese el costo de una fábrica donde se requieran de 50 a 100 servicios telefónicos [1].

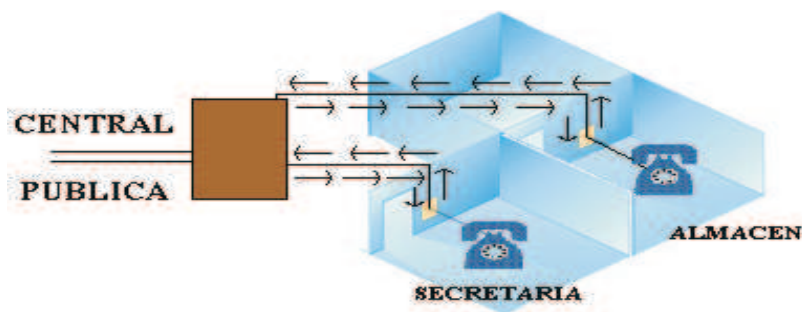


Figura 2.1: Llamada por central pública

Para evitar problemas de tal magnitud, se realizaron diversas implementaciones las cuales se describen cronológicamente.

2.2.2. Intercomunicador en paralelo

El primer implemento para resolver el problema de comunicarse telefónicamente dentro de una empresa a través de la central publica, es el intercomunicador en paralelo, que no es más que un simple aparato que permite comunicarse desde diferentes puntos en un mismo edificio, comunidad o empresa, el intercomunicador tenía como inconveniente que para hacer una llamada a una persona en específico se debían diseñar códigos.

Como por ejemplo: para el primer abonado un timbre, para el segundo dos, en el tercero tres y así sucesivamente, como se puede observar en la Figura 2.2, estos códigos hacían más complicada la comunicación conforme se aumentaban usuarios.

Otro problema es que la comunicación no era privada ya que cualquier otro aparato que se descolgara escucharía la conversación aunque permitía una conferencia por medio de los aparatos [1].

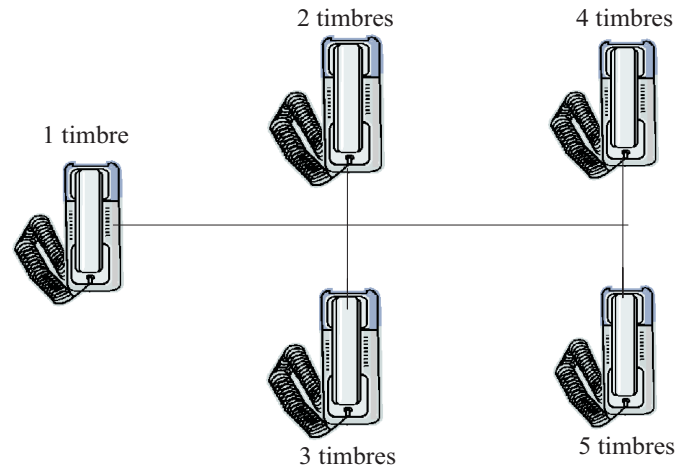


Figura 2.2: Intercomunicador en paralelo

2.2.3. Intercomunicador con señalización selectiva

Después vienen los intercomunicadores con señalización selectiva que permiten llamar a una persona en específico, sin la necesidad de diseñar códigos, simplemente se marca la extensión de la persona con la que se desea hablar y descolgando se entabla la comunicación claro, con la ventaja de que se podía hacer una conferencia entre varias extensiones y la desventaja de que no se podía tener una comunicación privada como en el intercomunicador en paralelo.

La Figura 2.3 se observa la distribución en un intercomunicador con señalización selectiva.

En la actualidad los equipos de señalización selectiva se usan ya no sólo como intercomunicación sino como porteros y abre puertas a distancia, con una amplia difusión en el mundo moderno como intercomunicadores [1].

Posteriormente se hicieron nuevas implementaciones surgiendo así los circuitos jefe-secretaria.

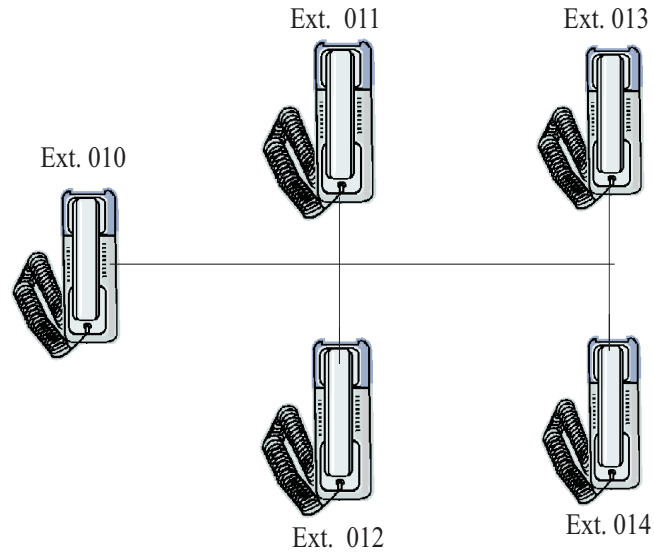


Figura 2.3: Señalización selectiva

2.2.4. Circuito Jefe-Secretaria

La implementación de circuitos jefe-secretaria, con teléfonos a manos libres, consistían en un juego de teléfonos los cuales permitían una comunicación punto a punto sin la necesidad de un micro teléfono y se activaba por medio de un botón el cual abría el circuito contrario y permitía escuchar la voz de la persona con la que se quería tener comunicación, ya sea el jefe o la secretaria; en la Figura 2.4 se muestra este proceso [1]. En la actualidad los circuitos jefe-secretaria, incorporaron un circuito para el manejo de

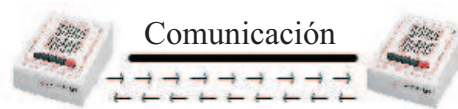


Figura 2.4: Circuito jefe-secretaria

una línea de abonado, que consiste en un pequeño circuito de retención, que permite la privacidad en la llamada ya sea interna-interna o interna-externa, son conocidos como juegos de teléfonos secretariales; la Figura 2.5 muestra la funcionalidad de los teléfonos secretariales [1].

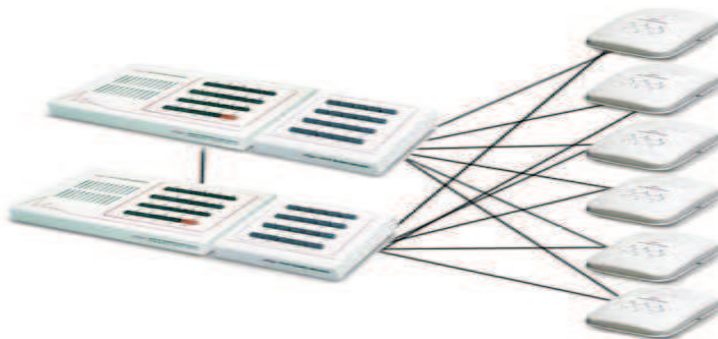


Figura 2.5: Teléfonos secretariales

La evolución del circuito jefe-secretaria dió paso a las pequeñas centrales telefónicas, que son circuitos conmutadores que permiten la comunicación interna de una empresa sea cual sea el giro o profesión. Con el tiempo, a estas centrales telefónicas, se les agregó circuitos de abonado de las centrales públicas.

2.2.5. Primeros PBX (Private Branch eXchange)

La aparición de las centrales telefónicas con los circuitos de abonado dio paso a los pequeños conmutadores privados que permitieron la administración de la comunicación de una empresa.

La administración de los recursos de comunicación se da en cuanto al enlace entre extensiones (comunicación interna) y las extensiones con troncales o troncales y extensiones (comunicación externa saliente y entrante). Permitiendo la conmutación para la correcta conexión y permitiendo privacidad en las comunicaciones.

Conforme la tecnología avanzó las centrales fueron mejoradas en funciones y capacidades para hacerlas más complejas y variadas, en la Figura 2.6 se muestra a un PBX [1].

2.3. ¿Que es un PBX?

Son las siglas de Private Branch Exchange. Los conmutadores sirven para interconectar a las personas de una organización entre ellas y con otras personas externas, optimizando

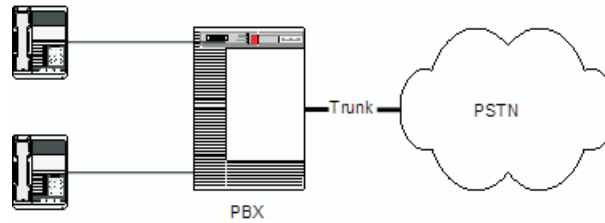


Figura 2.6: PBX private branch exchange

el uso de las líneas telefónicas y proporcionando ciertas facilidades a los usuarios tales como conferencia entre 3 o más personas, transferencia, desvío de llamada, entre otras operaciones.[2].

Los principales beneficios que puede brindar un PBX son lo siguientes:

- Menor costo de comunicación.
- Mejora la imagen del lugar de ubicación.
- Comunicación más fluida.
- Mayor productividad del personal.
- Mayor control sobre el gasto telefónico.

Dado a que no todos en una organización utilizan el teléfono simultáneamente, es posible compartir el número más limitado de líneas telefónicas sin que por ello se saturen. Un conmutador nos hace más productivos al inyectar fluidez en la realización de nuestras tareas. Además permite ejercer un control de llamadas para administrar presupuestos telefónicos y evitar malos usos.

Los conmutadores PBX son la base para un sin numero de servicios adicionales tales como correo de voz, centro de atención a clientes, etc.[2].

Existen tres tipos principales de sistemas que pueden ser confundidos con un PBX, aunque prácticamente se consideran como los sucesores de los PBX, tales sistemas se describen en las siguientes líneas [4].

1.-Private Automatic Branch Exchange (PABX).

Es un conmutador telefónico localizado en las instalaciones de la empresa, que transfiere llamadas entre las líneas internas y, entre líneas internas y líneas externas (PSTN). En contraste con el PBX, un PABX puede enrutar llamadas sin intervención humana, basándose solamente en el número de marcado. Sin embargo, no todos los PABXs pueden enrutar automáticamente llamadas externas a números internos [3].

2.-Private Automatic Exchange (PCX).

Son la evolución de alta capacidad del PBX digital (Digital Private Branch eXchange), que incorpora un sistema de conmutación basado en células o interfaces asíncronas (ATM), una arquitectura abierta cliente/servidor y una gran dependencia en la integración de la telefonía y la computación (CTI) para entregar funcionalidad avanzada, al tiempo que conserva la inversión de la empresa en tarjetas de línea PBX y equipos terminales [3].

3.-Computerized Branch Exchange (CBX).

Homónimo de PBX pero computadorizado. El sucesor de un PABX (Private Automatic Branch Exchange), los sistemas CBX son capaces de soportar mas de 600 usuarios internos. Solo es necesario agregar equipos de red IP para conectar la cantidad de teléfonos que sean necesarios [3].

Es posible configurar grupos de usuarios internos, los cuales pueden ser configurados para que todos suenen al mismo tiempo, o bien en secuencia, dada una llamada entrante para, por ejemplo, un departamento de la oficina determinado[3].

Desde los PBX hasta los CBX dependen de una unidad central de procesos mejor conocida como CPU.

2.3.1. CPU (Central Process Unity)

El sistema encargado de establecer las conexiones, llamado CPU controla, mediante un programa, las direcciones que debe tomar una llamada, la mejor ruta para la conexión, además de la facturación.

Esas funciones son muy sencillas con métodos computarizados, pero si se trata de

sistemas electromecánicos se convierte en una tarea bastante difícil, por suerte para las telecomunicaciones esos sistemas electromecánicos son hoy dispositivos obsoletos.

Algunas de las funciones que están disponibles en una PBX son:

- Transferencia de llamadas.
- Sistema para conocer el estado de las extensiones.
- Sistema de espera: Hace que si alguien llama a una extensión ocupada, el sistema haga esperar al realizador de la llamada hasta que la extensión quede libre.
- Conferencias, permite que llamadas del exterior lleguen a hablar con varias extensiones a la vez.
- Mantener un archivo con información sobre las comunicaciones.
- Sistema de contraseñas.
- Desviar llamadas a petición de los usuarios, por si se van a mover de su puesto.

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 que se realice la comunicación entre usuarios se establece un número de teléfono que acepta las llamadas entrantes, y tras pedir un código (o sin pedir código), pide amablemente que presionemos la extensión a la cual queremos llamar.

Desde dentro de la PBX, también se pueden realizar llamadas al exterior. Hace tiempo, existían empresas que tenían sus PBX totalmente públicas, al pulsar 9 se accedía a la línea externa, hasta que se dieron cuenta de que algunas personas que no eran de su empresa llamaban desde el PBX.

En una PBX habrá como mínimo una línea externa, que se coloca para que los usuarios puedan comunicarse con el exterior. Y también, como mínimo, habrá una línea desde el exterior, para que el exterior pueda comunicarse con los usuarios [1].

2.3.2. PBX más modernas

Las PBX modernas combinan un ordenador, un sistema de almacenamiento masivo de datos, y un sistema de conmutación de líneas. Esto nos permite:

- Producir un archivo de facturas detalladas para identificar y manejar las llamadas.
- Combinar los circuitos para comunicaciones de voz (día) y comunicaciones rápidas de datos (noches).
- Manejar correo electrónico.

La capacidad de comunicaciones de una PBX ha de ser cuidadosamente estudiada, porque ha de soportar las comunicaciones entre extensiones. Un factor crítico es el número de líneas y enlaces del sistema [5].

2.3.3. Capacidad de la PBX

La capacidad de un PBX se puede medir de dos formas: por el número de líneas telefónicas que puede atender, y por el número de extensiones que soporta. Los hay desde configuraciones mínimas, tal vez tres troncales (líneas) y ocho extensiones, hasta gigantes que soportan corporaciones geográficamente distribuidas, y con miles de empleados compartiendo un mismo sistema.

Así, a la hora de comprar uno, las empresas no sólo deben considerar que el conmutador cuente con una capacidad adecuada a sus requerimientos, sino también que pueda crecer en caso necesario.

Típicamente se trata de equipos modulares, de modo que, al ir adicionándoseles tarjetas, va aumentando el número de líneas y extensiones que manejan [5].

2.3.4. Funciones especiales

Además del ya mencionado detalle de la capacidad, es necesario definir otros factores para tomar en cuenta en un PBX: qué tipo de líneas se tiene o se piensa contratar con la

empresa telefónica (analógicas, digitales, número de grupo, red digital integrada y red digital de servicios integrados, entre otros), servicios de valor agregado que se desean, contratos de mantenimiento y soporte, etc.

Porque los conmutadores no sólo son llamadas. También suelen incluir (y cada vez más) funciones y características especiales tales como [6]:

- Operadora automática. Sistema que emplea grabaciones por el lado del PBX, e instrucciones por teclado desde el teléfono remoto, para que quien llama pueda dirigirse a la extensión que desea.
- Servicio de Respuesta interactiva. También combinando instrucciones de voz por parte del conmutador, y de teclado por parte de quien llama, permite que el usuario realice operaciones más complejas, como consultas de saldos, movimientos, recepción de fax en demanda, etc.
- Correo de voz. Cada persona tiene un buzón, en el cual la gente que llama puede dejar un mensaje en caso de ausencia o cuando está ocupada la extensión. Los fabricantes ofrecen el correo de voz como valor agregado, e incluso terceros proveedores las desarrollan para que los usuarios las monten en sus PBX.

2.3.5. Recomendaciones para la administración de un PBX

Las PBX y su economía

Al comprar un PBX, las empresas están haciendo una inversión que por definición acabará pagándose, pues por el hecho de poder compartirlas, automáticamente se reduce el número de líneas telefónicas a contratar.

Si se hace el cálculo de lo que se estará ahorrando por concepto de contratación y recibos telefónicos, se verá que la cifra no es en absoluto despreciable. Un factor menos medible, pero a todas luces más significativo, es la ganancia en productividad que se logra con mayor eficiencia en las comunicaciones telefónicas.

Una empresa de dos o tres pisos, que en vez de un conmutador tiene líneas telefónicas separadas. Además de que se necesitarán más líneas para llegar a todos los empleados,

tener tantas líneas presenta dos desventajas evidentes.

La desventaja más evidente será un problema de líneas ocupadas. Si en el área de Ventas (o Soporte técnico, Cobranza o la que se prefiera) hay una línea, y alguien descuelga, estará bloqueando las comunicaciones tanto de entrada (quien llame se topará con el tono de ocupado) como de salida (sus compañeros tampoco podrán llamar, aun cuando en otros departamentos tal vez no estén usando las líneas).

Al permitir que todas las líneas estén disponibles para todos los empleados, independientemente de la ubicación de éstos, el conmutador elimina tal desperdicio de recursos. Entonces contar con un PBX tiene una ventaja adicional, que es la comunicación interna. Porque, además de enlazar líneas externas con extensiones internas, estos aparatos también funcionan como un sistema de comunicaciones intraempresarial.

Y esa eficiencia en las comunicaciones no puede significar otra cosa sino mayor productividad dentro de una organización.

¿Cómo se conforma la oferta de conmutadores disponible en el mercado? Algunos de los fabricantes más importantes son, en estricto orden alfabético: Alcatel, Ericsson, Harris, Lucent Technologies, NEC, Nortel Networks, Panasonic, Siemens, Tadiran y Telrad. Cada uno de ellos maneja diversos modelos, distintos en capacidad y funcionalidades [5].

Capítulo 3

Telefonía IP y VoIP

3.1. Introducción

La telefonía al igual que el resto de la tecnología de información está caminando rápidamente hacia los sistemas abiertos y portables compatibles con diferentes marcas de equipo y manejando estándares comunes que permitan la integración y el crecimiento.

Las actuales capacidades de cómputo han hecho de la voz una forma de información digital procesable a través de equipo de cómputo y aplicaciones especializadas es por esta razón que en el presente capítulo se explicara detalladamente las generalidades de la telefonía IP y de VoIP.

3.2. La telefonía IP en la actualidad

Actualmente todas las grandes compañías telefónicas tradicionales están migrando sus infraestructuras de voz hacia la telefonía IP por el hecho de que es más económico tener y mantener una sola infraestructura para voz y datos [7].

Cuando se toma la decisión de migrar toda la telefonía de una empresa a telefonía IP ya no se tendrá que invertir en un cableado para datos y otro para telefonía, de esta forma las computadoras, servidores, impresoras y teléfonos funcionan bajo la misma red, porque además la telefonía demanda pocos recursos, incluso en redes saturadas de

flujo de información la telefonía IP opera eficientemente [7].

3.3. Sistema tradicional de telefonía

El servicio telefónico es, junto con la red eléctrica, uno de los más confiables que se conocen y se usan, ya que todo es redundante y está pensado para funcionar constante y permanentemente. Una central telefónica esta diseñada para minimizar los tiempos de interrupción del servicio [8].

En general todos tienen un teléfono en casa consecuentemente el cable del teléfono tiene una roseta (RJ-11) parecida a la del cable de red, y que dentro tiene dos cables de cobre, al que se denomina par telefónico, el cual va hasta la central telefónica, a una placa que se la suele denominar placa de abonado. Es la placa que controla nuestra línea [8].

En realidad, puede controlar muchas líneas, no una sola, y tiene una densidad de puertos que depende del fabricante, oscila entre los 8 y 16 abonados (a veces más, a veces menos), el valor exacto depende del equipo en particular. La central telefónica es un conjunto de equipos relacionados. Todo este conjunto forma un equipo muy grande que puede llegar a ocupar varias habitaciones [8].

Las centrales telefónicas suelen estar diseñadas para tener alta disponibilidad (se acostumbra a decir que son carrier class, dado que se dice están disponibles en un porcentaje del 99.9 del tiempo, que representa alrededor de 5 minutos al año de interrupción de servicio)[8].

Para lograr este objetivo, cuentan con redundancia en múltiples niveles (procesadores, enlaces, etc.); y en general se conectan a un sistema de energía interrumpida, que tiene un número considerable de baterías que se conectan a un grupo electrógeno que se activa cuando se corta la luz [8].

3.3.1. Procesamiento de llamadas

La voz llegaba hasta la central de forma analógica, actualmente ya no es común utilizar centrales analógicas, todo el proceso que hay desde que llega la señal a una central y

sale de otra hacia el otro abonado, es digital [7].

La placa de abonado es la que se encarga de hacer la conversión de una señal analógica a una digital y viceversa. La señal se convierte a un PCM de 64kbps, que es una señal digital sin pérdida de información y sin compresión, este formato es el que se está utilizando desde sus comienzos. También es la placa de abonado la que decodifica los tonos de discado (DTMF). Es decir que, se utiliza el concepto de señalización en banda: comandar a la central utilizando la misma banda por la que se habla [7].

3.3.2. Conexión entre centrales

La llamada que sale de la central tiene que llegar hasta la central donde está la persona con la que se quiere hablar. No hay doscientos millones de cables entre una y otra, sino que hay un enlace, el cual puede ser de diversos tipos. Este enlace se debe multiplexar para que todos los abonados de la central puedan hablar por teléfono [9].

Esta multiplexación es la que hace la diferencia a la hora de la calidad del servicio para el usuario. El sistema de multiplexación que utilizan las centrales telefónicas se llama TDM, consiste en dividir el flujo de datos en partes iguales de tiempo a 64kbps (llamadas time-slots), de manera que los datos correspondientes al primer abonado van en el primer time-slot, los correspondientes al segundo en el segundo, y así sucesivamente [9].

Suponiendo un enlace de 2 Mbps de ancho de banda en el que se transmiten 64kbits, podría haber hasta 32 abonados hablando a la vez. Con esta multiplexación en tiempo, se separan y luego vuelven a unir los flujos de voz que van de una central a otra, de manera transparente para el que lo está utilizando [9].

La particularidad de esta tecnología es que como se divide por un tiempo fijo, se puede garantizar el time-slot y saber que siempre lo que corresponde al primer abonado va en el primer time-slot, una vez establecida la comunicación, sea de un lugar a una cuadra de distancia o de algún país a otro país, está garantizado el ancho de banda necesario para poder hablar sin interrupciones [9].

3.3.3. Ruteo, señalización y protocolos

Un punto importante es el ruteo entre centrales, es decir, ¿cómo sabe la central del abonado con qué central se tiene que conectar?. Se denomina señalización a la información relacionada con una llamada que se transmite entre dos equipos, se puede dividir en dos grupos: la que refiere al abonado y las llamadas en sí (se levanta, se marca, se cuelga), y otra parte entre las centrales [9].

A través de la señalización, la central puede ubicar a qué otra central tiene que llamar, a qué abonado dentro de esa central hay que llamar, saber que si se cortó la comunicación, o que dio tono de ocupado, etc. Las centrales entre sí se comunican utilizando diversos protocolos, los cuales generalmente son estándares públicos, aunque en muchos casos las especificaciones no son fáciles de conseguir. Los protocolos más comunes son tres: R2, PRI y SS7 [9].

Se necesita que las dos centrales que intentan comunicarse entre si puedan hablar un mismo protocolo, de manera que si se quieren intercomunicar dos centrales que no soportan los mismos protocolos, es necesario que utilicen una central intermedia que traduzca la información [9].

Acerca del enlace por el cual se pasa tanto la señalización como la voz en sí, existen diversos tipos. Los más conocidos y comunes son E1 o E3 (europeos), con sus variantes T1 o T3 (utilizadas principalmente en los Estados Unidos). Son cables de cobre, muy parecidos al cable coaxial, que pueden ser de 75 o 120 ohms. El E1 tiene 2Mbps (32 canales de 64kbps), el E3 tiene 32Mbps (512 canales de 64kbps) [9].

En muchos ámbitos cuando se habla de enlaces E1 ó E3 se le da importancia sólo al ancho de banda; sin embargo para telefonía IP también interesa el número de times-lots en el cual se puede dividir, sin embargo, no se pueden ocupar todos los canales para pasar todos los abonados [9].

Por ejemplo, en el caso de una E1 se suelen utilizar 30 canales para el paso de la voz, 1 para el framing (el 0) y 1 para la señalización (el 15). En el de framing se suele encontrar el CRC de los otros 31 (aunque depende de la configuración), de manera que si un determinado frame se encuentra dañado, se le puede identificar y actuar inmediatamente [9].

Finalmente se puede decir que en los sistemas telefónicos tradicionales existe un conmutador con una tarjeta de extensión para cada usuario y una tarjeta de línea para cada troncal contratada con la compañía telefónica [9].

El problema es el crecimiento, ya que para cada nuevo usuario o cada nueva línea hay que comprar tarjetas de expansión que solo el fabricante del conmutador vende y quizás para cuando se requiere la expansión, el modelo del conmutador ya esta descontinuado y las expansiones son aún mas complicadas de adquirir [9]. Adicionalmente cada extensión requiere de un cable desde el conmutador hasta la extensión haciendo complicada y costosa la instalación [9].

Un ejemplo del sistema tradicional de telefonía se puerder ver en la Figura 3.1

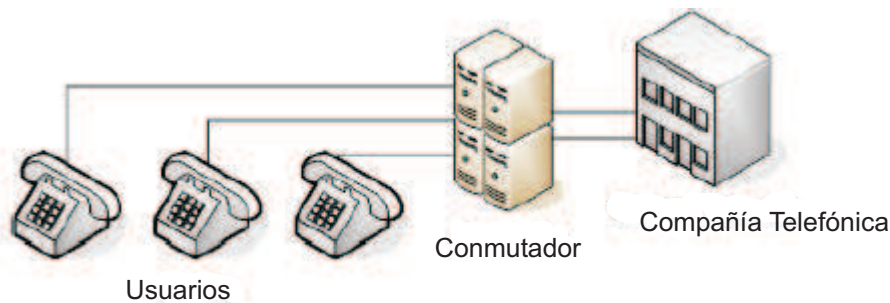


Figura 3.1: Sistema tradicional

3.4. Telefonía IP

La Telefonía IP es una tecnología que permite la transmisión de la voz a través de redes IP en forma de paquetes de datos. La Voz sobre IP es una aplicación de la telefonía IP, de forma que permite la realización de llamadas telefónicas ordinarias sobre redes IP u otras redes de paquetes utilizando una PC, gateways y teléfonos estándares[10].

En general, los servicios de comunicación-voz, fax, aplicaciones de mensajes de voz, son transportados normalmente vía redes IP en Internet, en lugar de ser transportados vía la red telefónica convencional [10].

3.4.1. Codificación de la voz

La transmisión ya no se hace en PCM, como en la telefonía tradicional, la voz se puede comprimir: si una persona se queda callada, por ejemplo, no es necesario transmitir el sonido completo del silencio. Hay muchos codecs de compresión, como todo codec, cuanto más se comprime, más procesador se necesita, existen codecs con pérdida que comprimen de 64kbits a 4kbits, incluso hasta 3.1kbits, ó algunos que son sin pérdida, pero la mayoría son con pérdida [10].

Se han realizado diversos estudios al respecto, ya que lo más importante es la percepción que tiene la gente de lo que se escucha, y es altamente difícil medir la percepción humana. Para la realización de estos estudios, se comprime el audio y se pide a grupos de personas que lo escuchen y que manifiesten si les parece que es de buena calidad o no, por ultimo se les asignan puntajes [10].

En general se elige un balance entre compresión y percepción, se consideran distintos balances. Existen muchos codecs que están patentados, para los que hay que pagar las licencias de uso [10].

Un ejemplo de un codec es el GSM, utilizado en los teléfonos celulares. Es un codec libre, que se escucha bastante bien, comprime bastante bien, y consume muy poco procesador. Que consuma poco procesador es importante cuando se está trabajando a gran escala (200, 1000 líneas), en el caso de los celulares, la voz se comprime en el mismo aparato celular y se transmite ya comprimida. Para este protocolo, en GNU/Linux existe la libgsm que es una biblioteca pequeña y útil [10].

3.4.2. Señalización

Es necesario tener un protocolo de señalización para poder indicar a qué máquina se quiere llamar y demás. Existen actualmente varios protocolos para señalización, uno que se esta dejando de utilizar es el H.323 pero es muy importante porque fue el primero que se empezó a usar en VoIP de forma masiva, probablemente en el futuro no se use más. Los programas NetMeeting, y su equivalente libre GnomeMeeting utilizan este protocolo [10].

El protocolo que más se está usando actualmente es SIP: Session Initiation Protocol. Se trata de un protocolo que tiene una característica muy particular: está estandarizado por la IETF y, en consecuencia, es muy abierto y de fácil acceso.

SIP es un protocolo de texto plano que se utiliza sobre TCP, ya que en el caso de la señalización es importante que no se pierda la información. Tiene una arquitectura que está muy bien pensada, sin embargo, tiene problemas para atravesar NAT. Normalmente, cuando se usa SIP, el protocolo que se utiliza para enviar la voz es RTP, que se usa sobre UDP. [10].

3.4.3. ¿Como funciona la telefonía IP?

La Telefonía sobre IP convierte las señales de voz estándar en paquetes de datos comprimidos que son transportados a través de redes de datos en lugar de líneas telefónicas tradicionales. La evolución de la transmisión conmutada por circuitos a la transmisión basada en paquetes toma el tráfico de la red pública telefónica y lo coloca en redes IP bien aprovisionadas. Las señales de voz se encapsulan en paquetes IP que pueden transportarse como IP nativo o como IP por Ethernet, Frame Relay, ATM o SONET [11].

En los sistemas telefónicos IP existe una sola infraestructura de red tanto para las aplicaciones de voz como para las aplicaciones de datos, con un solo cableado las computadoras y los teléfonos se interconectan y comunican; en la Figura 3.2 se observa un ejemplo de la interconexión de voz y datos [11].

Regular y recomendablemente es que el conmutador debe ser un PC/Servidor mínimo a 1.8 Ghz con 40 GB de disco y 256 KB de memoria corriendo el sistema operativo Linux que le da una altísima estabilidad y redundancia y las aplicaciones del conmutador [11].

Para conectar las líneas telefónicas convencionales al servidor se le instalan unas tarjetas PCI que pueden manejar una o varias líneas simultáneamente, estas tarjetas son fabricadas por diferentes empresas de forma tal que los costos de Hardware siempre van a la baja.[11]

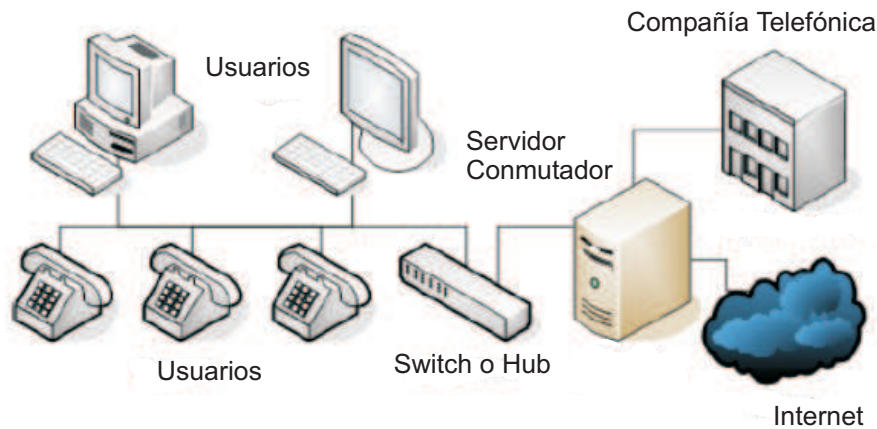


Figura 3.2: Sistema de telefonía IP

3.5. La telefonía IP y las telecomunicaciones

Frente al constante cambio de las telecomunicaciones, la telefonía sobre IP es excepcionalmente prometedora. Ante un mercado global cada vez más competitivo, las compañías telefónicas ya existentes, los proveedores de servicios de Internet buscan, en forma constante, maneras de aumentar sus ofertas de servicios [5].

La telefonía sobre IP ha captado la atención de dichos proveedores de servicios en todo el mundo, ofreciendo una amplia gama de servicios nuevos y reduciendo al mismo tiempo sus costos de infraestructura. La voz sobre IP está cambiando la perspectiva del acceso a la información, fusionando voz, datos, y funciones multimedia en una sola infraestructura de acceso convergente [5].

Mediante la telefonía sobre IP, los proveedores pueden ofrecer servicios de voz básicos y ampliados a través de Internet, incluyendo la llamada en espera en Internet, el comercio en la web por telefonía ampliada y comunicaciones interactivas de multimedia. Los servicios ofrecidos se integrarán de manera interrumpida a las redes conmutadas existentes (PSTN) a fin de permitir que se originen o terminen llamadas en teléfonos tradicionales según sea necesario [5].

Dado que IP es una norma abierta, VoIP le brinda a los proveedores de servicios, flexibilidad para personalizar sus servicios existentes e implementar nuevos servicios con mayor rapidez y eficiencia en función de los costos de antes, incluso en áreas remotas

dentro de su región [5].

3.6. Tecnología VoIP

La VoIP inicialmente se implementó para reducir el ancho de banda mediante compresión vocal, aprovechando los procesos de compresión diseñados para sistemas celulares en la década de los años 80. En consecuencia, se logró reducir los costos en el transporte internacional. Luego tuvo aplicaciones en la red de servicios integrados sobre la LAN e Internet. Con posterioridad se migró de la LAN a la WAN [10].

VoIP designa a la tecnología empleada para enviar información de voz en forma digital en paquetes discretos a través de los protocolos de Internet, en vez de hacerlo a través de la red de telefonía habitual [10].

Antes de seguir, tal vez sea conveniente aclarar qué es un protocolo de conexión. Un protocolo de conexión es un conjunto de normas, un "lenguaje en común" que ambas partes acuerdan utilizar para poder comunicarse, es como decir: Ahora vamos a comunicarnos en inglés, y nos ponemos de acuerdo en que esto es inglés, o sea es una convención [10].

La industria de Voz sobre IP se encuentra en una etapa de crecimiento rápido. La evolución del uso de Voz sobre IP vendrá con la evolución de la infraestructura y de los protocolos de comunicación. En el año 2010, una cuarta parte de las llamadas mundiales se basarán en IP [10].

A lo largo del tiempo, las aplicaciones de voz y datos han requerido redes distintas que usan tecnologías diferentes. Sin embargo, últimamente se han realizado numerosos esfuerzos para encontrar una solución que proporcione un soporte satisfactorio para ambos tipos de transmisión sobre una sola red [10].

La Voz sobre IP, como se muestra en Figura 3.3 es una tecnología de telefonía que puede ser habilitada a través de una red de datos de conmutación de paquetes, vía el protocolo IP. La ventaja real de esta tecnología es la transmisión de voz de forma libre, ya que viaja como datos [10].

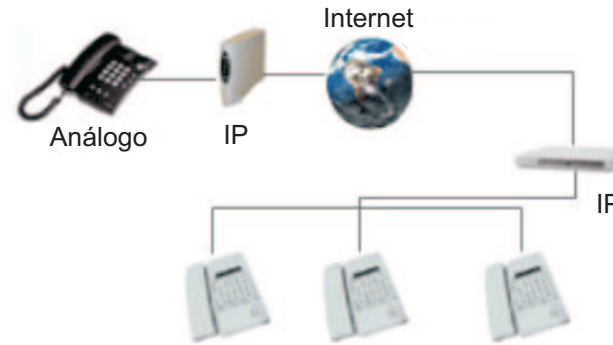


Figura 3.3: Tecnología VoIP

La tecnología VoIP puede revolucionar las comunicaciones internas al ofrecer:

- Acceso a las redes corporativas desde pequeñas sedes a través de redes integradas de voz y datos conectadas a sucursales.
- Directorios corporativos basados en la Intranet con servicios de mensajes y números personales para quienes deben desplazarse.
- Servicios de directorio y de conferencias basadas en gráficos desde el sistema de sobremesa.
- Redes privadas y gateways virtuales gestionados para voz que sustituyen a las VPN.

VoIP brinda nuevas oportunidades para quienes sean capaces de preverlas y actúen con la rapidez suficiente para superar la confusión que envuelve esta extraordinaria tecnología.

Hay dos posibilidades de conexión:

- Una de las partes tiene VoIP y la otra no.
- Ambas partes tienen VoIP.

Si ambas partes tienen VoIP la llamada es totalmente gratuita, pues se llama de VoIP a VoIP tiene que discar el número telefónico y nada más, Si sólo quien llama tiene VoIP

, entonces hace uso de una tarjeta que se compra online. La mencionada tarjeta no es una tarjeta de plástico o de cartón como las que se venden en los comercios, mas bien es una tarjeta virtual que se compra y carga por Internet [10].

Es necesario aclarar que se puede instalar un VoIP aunque tenga una central telefónica y más de una línea de teléfono, pues se puede designar una línea para que trabaje directamente con el VoIP, sin perjuicio de seguir utilizándola normalmente. La VoIP es una buena alternativa para quien tiene oficinas en el exterior y hace llamadas de larga distancia diariamente o de mucha duración [10].

3.6.1. Elementos de VoIP

El modelo de Voz sobre IP está formado por tres principales elementos:

- El cliente. El cliente establece y termina las llamadas de voz. Codifica, empaqueta y transmite la información de salida generada por el micrófono del usuario. Asimismo, recibe, decodifica y reproduce la información de voz de entrada a través de los altavoces o audífonos del usuario [10].

Cabe destacar que el elemento cliente se presenta en dos formas básicas: la primera es una suite de software corriendo en una PC que el usuario controla mediante una interfase gráfica (GUI); y la segunda puede ser un cliente "virtual" que reside en el gateway [10].

- Servidores. El segundo elemento de la voz sobre IP está basado en servidores, los cuales manejan un amplio rango de operaciones complejas de bases de datos, tanto en tiempo real como fuera de él [10].

Las operaciones incluyen validación de usuarios, tasación, contabilidad, tarificación, recolección, distribución de utilidades, enrutamiento, administración general del servicio, carga de clientes, control del servicio, registro de usuarios y servicios de directorio entre otros.

- Gateways. El tercer elemento lo conforman los gateways de Voz sobre IP, que proporcionan un puente de comunicación entre los usuarios. La función principal de un gateway es proveer las interfases con la telefonía tradicional apropiada, funcionando como una plataforma para los clientes virtuales.

Los Gateways también juegan un papel importante en la seguridad de acceso, la contabilidad, el control de calidad del servicio (QoS) y en el mejoramiento del mismo [10].

3.6.2. Características de VoIP

Por su estructura el estándar proporciona las siguientes características:

- Permite el control del tráfico de la red, por lo que se disminuyen las posibilidades de que se produzcan caídas importantes en el rendimiento de las redes de datos [10].
- Proporciona el enlace a la red telefónica tradicional [10].

Al tratarse de una tecnología soportada en IP presenta las siguientes ventajas adicionales:

- Es independiente del tipo de red física que lo soporta. Permite la integración con las grandes redes de IP actuales [10].
- Es independiente del hardware utilizado [10].
- Permite ser implementado tanto en software como en hardware, con la particularidad de que el hardware supondría eliminar el impacto inicial para el usuario común [10].

3.6.3. Protocolos de VoIP

Hoy en día, existen dos protocolos para transmitir voz sobre IP, ambos definen la manera en que los dispositivos de este tipo deben establecer comunicación entre sí, además de incluir especificaciones para codecs (codificador-decodificador) de audio para convertir una señal auditiva a una digitalizada compresada y viceversa [10].

H.323

H.323 es el estándar creado por la Unión Internacional de Telecomunicaciones que se compone por un protocolo sumamente complejo y extenso, el cual además de incluir la voz sobre IP, ofrece especificaciones para vídeo-conferencias y aplicaciones en tiempo real, entre otras variantes [10].

SIP

Session Initiation Protocol (SIP) fue desarrollado por la IETF específicamente para telefonía IP, que a su vez toma ventaja de otros protocolos existentes para manejar parte del proceso de conversión, situación que no se aplica en H.323 ya que define sus propios protocolos bases [10].

3.6.4. Aplicaciones de VoIP

VoIP proporcionaría a los departamentos de una misma empresa, comunicaciones gratuitas entre si, con el ahorro de costos que esto supondría, no solo entre sus sucursales, sino entre proveedores, intermediarios y vendedores finales, las comunicaciones se podrían realizar de forma completamente gratuita [10].

Además, la red de comunicaciones de la empresa se vería enormemente simplificada, ya que no habría que cablear por duplicado la red, debido a que se aprovecharía la red de datos para voz [10].

Conexión de 2 PCs

A modo de ejemplo, si se consideran dos PCs, que están conectadas a través de Internet. Juan, que está conectado desde una PC, quiere hablar con María, que está conectada desde otra. A María le llega un invite que le indica que Juan quiere hablar con ella (equivalente a un RING), y si acepta la comunicación (equivalente a levantar el teléfono), puede hablar con Juan [10].

La conexión se establece usando SIP sobre TCP y luego la transmisión se hace usando

RTP sobre UDP. Cuando se termina la conversación, por SIP se transmite la finalización de la conexión, esto permite que dos usuarios de PC puedan hablar por teléfono, sin tener una central telefónica de por medio, utilizando la estructura IP existente para establecer una comunicación. Durante la inicialización se pasan las IPs y los puertos a utilizar y por eso es que es difícil hacerlo a través de NAT [10].

Conexión de muchas computadoras

Si en lugar de 2 PCs, se quiere conectar un número importante de computadoras, que quieren hablar entre sí sin tener que estar transmitiéndose los números de IP, y el que les está proveyendo el servicio quiere tener un registro de las comunicaciones establecidas, se utiliza un Server SIP (que vendría a ser el equivalente a un Gatekeeper en H.323) [10].

También se le puede llamar Proxy SIP o Router SIP, que si bien teóricamente cumplen funciones específicas, en general se utilizan los términos de manera indistinta. Teniendo un server, cuando Juan quiere hablar por teléfono, le envía una señal al server indicándole que quiere hablar con María, y este le avisa a María que Juan quiere hablar con ella.

A partir de que se acepta la comunicación, se pasan algunos mensajes más a través del server (utilizando SIP) para negociar IPs, puertos, protocolo de compresión a utilizar, etc. Pero una vez que comienza la comunicación, el canal UDP ya no pasa por el server. Una vez terminada la conversación, se utiliza SIP para avisar que se terminó la conversación [10].

Esta es una de las mejores cosas que tiene la telefonía IP, porque por un lado separa la señalización de la transmisión de voz, y por el otro lado la transmisión se hace peer to peer. Pero trae consigo que el server debe confiar en el correcto uso de los clientes para saber cuándo una comunicación se terminó realmente [10].

Un cliente que tenga DHCP tiene que avisarle al servidor SIP en qué IP está, para esto puede autenticarse contra él, utilizando un nombre de usuario y una clave. De manera que el servidor SIP puede saber que un determinado usuario no está y poner un contestador, dar ocupado [10].

Con este principio se puede hacer que un teléfono VoIP se conecte en cualquier lugar del mundo donde haya banda ancha y siempre sigue siendo el mismo teléfono. Y de hecho este servicio existe y se vende. Por ejemplo, si se obtiene una línea en la ciudad de México o en cualquier otro país, y se quiere llevar el teléfono VoIP a cualquier lugar del mundo, lo puede conectar a un ADSL y se puede hablar o recibir llamadas como si se estuviera en la ciudad de México [10].

De la misma manera que con las centrales telefónicas, puede haber varios servidores que se comuniquen entre sí, y solamente van a intercambiar la parte correspondiente al protocolo SIP, la parte de RDP/UDP se hace directo entre los dos puntos que se están comunicando. La implementación de referencia del server SIP es Open Source.

Por otro lado, se puede hablar desde una computadora a teléfonos comunes, para esto se necesita un gateway que haga la conversión de una tecnología a otra.[10]

Implementaciones

A nivel personal, por ejemplo se puede hablar con la familia que viva en algún lugar distante a través de VoIP, y otro día, hablar por teléfono de verdad, es una opción para ahorrar costos. Pero, cuando se habla de una implementación a nivel telefonía real (como la de las tarjetas para hacer llamadas a bajo costo) es diferente, tiene que ir por un enlace controlado [10].

Una posibilidad para tener una red de VoIP interna, por ejemplo, es tener unos auriculares y un micrófono en cada estación (PC). Y por otro lado, tener un equipo que se encarga de hacer la conversión a la telefonía tradicional. Para ello, es necesario tener algún contacto con la telefonía tradicional, una E1 ó líneas telefónicas [10].

En el caso de la E1, hará falta tener el hardware que se encargue de hacer TDM, para las líneas telefónicas, es necesario tener placas especiales que se conectan a líneas telefónicas [10].

3.7. Comparación de VoIP y la telefonía tradicional

VoIP es transmitir Voz utilizando IP. Si bien es una tecnología novedosa, tiene muchas características similares a las de la telefonía tradicional. Por eso, a continuación se explican brevemente las ventajas que existen entre la Voz sobre IP y la telefonía tradicional [11].

3.7.1. Ventajas

- Un único número de teléfono

Casi como un celular, si se tiene una conexión a internet en la oficina y en el hogar, el número telefónico será el mismo. Mejor aún, si llevamos una laptop a otro estado y la conectamos a la red, se tiene el mismo número de teléfono. Le permite tener un número de teléfono local que transfiera los llamados de sus familiares y amigos a cualquier parte del mundo que usted elija.

Así sus familiares y amigos podrán hablar con usted por sólo el costo de una llamada local mientras usted paga por el consumo de minutos [11].

- Ahorro en llamadas de larga distancia

Las mayores ventajas que va a ver un usuario en el hogar es la del ahorro en las llamadas de larga distancia ya que las comunicaciones no dependerán del tiempo en el aire.

Es decir no dependerá de la duración de la llamada, como se esta acostumbrados hasta ahora, sino más bien por el precio de mercado del proveedor de Internet, ya que se estará pagando por un servicio más dentro del paquete de datos que brinda la red [11].

- Llamadas a teléfonos fijos o celulares

Otra gran ventaja de la telefonía IP es que se puede llamar a un teléfono fijo o móvil en cualquier lugar del mundo para transmitir fax, voz, vídeo, correo electrónico por teléfono, mensajería y comercio electrónico.

Es decir, la gran variedad de servicios brindados por un solo operador es una de las grandes ventajas que ven los usuarios en el hogar y corporativos [11].

- Mensajería unificada y correo de voz

Cuando se está de viaje o fuera de casa u oficina, en vez de marcar el teléfono y clave para escuchar la lista de mensajes, hace la función de un sistema telefónico que le proporcione, en su computadora, un listado de esos mensajes y que le permita escucharlos y marcar teléfonos de su libro electrónico de direcciones con un simple click en su ratón [11].

La tecnología VoIP le permite realizar llamadas telefónicas y enviar faxes a través de una red de datos IP como si estuviese utilizando una red tradicional [11].

- Ventajas para las empresas

La convergencia de servicios de voz, datos y vídeo en una sola red para una empresa que lo adopte implica, un menor costo de capital, procedimientos simplificados de soporte y configuración de la red y una mayor integración de las ubicaciones remotas y oficinas sucursales en las instalaciones de la red corporativa [11].

La Telefonía IP utiliza la red de datos para proporcionar comunicaciones de voz a toda la empresa, a través de una sola red de voz y datos. Es evidente que el hecho de tener una red en vez de dos, es beneficioso para cualquier organización [11].

VoIP proporcionaría a las sucursales de una misma empresa, comunicaciones gratuitas entre ellas, con el ahorro de costos que esto supondría. No solo entre sus sucursales, sino entre proveedores, intermediarios y vendedores finales, las comunicaciones se podrían realizar de forma completamente gratuita [11].

Además, la red de comunicaciones de la empresa se vería enormemente simplificada, ya que no habría que cablear por duplicado la red, debido a que se aprovecharía la red de datos para voz. La capacidad permite a las compañías reducir los costos de fax y teléfono, agrupar los servicios de datos, voz, fax y vídeo, y construir nuevas infraestructuras de red para aplicaciones avanzadas de comercio electrónico [11].

- Centros de llamadas por la Web

Partiendo de un negocio que ofrece sus productos en línea, los visitantes de la Web no solo tendrán acceso a la información que la Web les proporciona, sino que además podrían establecer comunicación directa con una persona del departamento de ventas sin necesidad de cortar la conexión [11].

El poder tener contacto directo con alguien de la empresa de ventas reduciría el enorme temor del usuario a hacer sus compras por Internet por primera vez. Al establecer una conversación directa, le da una confianza que finalmente supondrá una mejora en su relación con el comercio electrónico [11].

- Videoconferencia integrada ó Multiconferencia

Con los datos de ancho de banda requeridos actualmente (de 8 a 16kbps por llamada), se podrían establecer de 15 a 30 comunicaciones simultáneas con una línea ADSL estándar, que podría satisfacer los requerimientos de una mediana empresa [11].

- Posibilidad de usar Push 2 Talk

De igual forma, con el simple gesto de pulsar un botón se establece comunicación directa con la persona que lo ha elaborado [11].

- Ventajas para los operadores o proveedores del servicio

Es obvio que este tipo de redes proporciona a los operadores una relación ingreso/recursos mayor, es decir, con la misma cantidad de inversión en infraestructura de red, obtienen mayores ingresos con las redes de conmutación de paquetes, pues puede prestar más servicio a sus clientes, otra posibilidad sería que prestará más calidad de servicio, velocidad de transmisión, por el mismo precio [11].

Capítulo 4

Protocolo de inicio de sesión (SIP)

4.1. Introducción

En el presente capítulo se explican a detalle las particularidades de SIP, este nuevo protocolo de señalización de medios de comunicación y mensajería instantánea que día a día gana terreno en el ámbito de la comunicación.

Al final del capítulo se podrá estructurar, modificar, y finalizar sesiones multimedia, también se dominarán las solicitudes y respuestas a peticiones de presencia y mensajes instantáneos sobre el Internet. Cabe mencionar que todas las imágenes de este capítulo fueron obtenidas del libro “Session Initiation Protocol (SIP) Extension for Instant Messaging” [24].

4.1.1. Antecedentes de SIP

El SIP fue desarrollado originalmente por el IETF. La Versión 1.0 de este protocolo, se presentó como un proyecto de Internet en 1997. Se hicieron cambios significativos al protocolo y se desarrolló una segunda versión, versión 2.0, que se presentó de igual forma como un proyecto de Internet en 1998.

El protocolo conseguido después de las modificaciones de versión se estandarizó en Marzo de 1999 y se publicó como RFC 2543 [12] en Abril de 1999. En septiembre de 1999, fue establecido por el IETF el grupo de trabajo del SIP para enfrentarse al

creciente interés en el protocolo.

A principios de julio del 2000 se presentó un proyecto de Internet que contenía correcciones a errores y aclaraciones para el SIP, se le llamo RFC 2543 "bis". Este documento fue eventualmente publicado como RFC 3261 [13], qué reemplazó a la especificación original RFC 2543.

La popularidad del SIP en el IETF permitió la formación de otros grupos de trabajo relacionados con este protocolo. El grupo de trabajo en la investigación de este protocolo fue formado para investigar las aplicaciones, desarrollar las condiciones necesarias para extensiones y publicar los mejores y más actualizados documentos prácticos acerca del uso del SIP.

Este grupo de trabajo también publica paquetes de RFCs acerca del SIP. El grupo de trabajo de SIP para la Mensajería Instantánea (SIMPLE) fue formado para estandarizar los protocolos relacionados con la presencia y aplicaciones de mensajería instantánea.

Para pasar de una simple propuesta a ser proyecto, un protocolo debe tener múltiples implementaciones de trabajo independientes y una experiencia operacional limitada. Desde los primeros días de la RFC 2543, los eventos de prueba de interoperabilidad SIP, llamados SIPit (aunque formalmente llamada "bakeoffs"), han sido cambiados unas pocas veces por año.

El nivel final, estandarizado, es archivado después de que su exitosa operatividad ha sido demostrada [14]. Con el documento de interoperabilidad de SIPit, el protocolo SIP puede modificar los estándares normales en el futuro.

SIP incorpora elementos de dos protocolos de Internet ampliamente utilizados: HTTP usado por navegadores Web y el SMTP usado por e-mail. De HTTP, SIP pidió prestado el diseño cliente-servidor y el uso de URLs y URIs. De SMTP, SIP pidió prestado un esquema de codificación de texto y el estilo de cabeceras de campo. Por ejemplo, SIP reutiliza las cabeceras de SMTP tales como TO, FROM, DATE y SUBJECT.

4.2. ¿Qué es SIP?

El Protocolo de Inicio de Sesión (SIP) es un nuevo protocolo de señalización de medios de comunicación y mensajería instantánea desarrollado para estructurar, modificar, y finalizar sesiones multimedia, solicitud y respuesta a peticiones de presencia y mensajes instantáneos sobre el Internet [15].

El SIP fue desarrollado por el IETF como una parte de la Arquitectura de Conferencias Multimedia en Internet y fue diseñado para unirse con otros protocolos de Internet como el TCP, TLS, UDP, IP, DNS, y otros.

Como su nombre lo implica, el protocolo SIP permite a dos puntos terminales establecer comunicación entre si a través de sesiones multimedia. Las principales funciones de señalización del protocolo son las siguientes:

- Localización de un punto terminal.
- Contactar a un punto terminal para determinar el establecimiento de una sesión.
- El intercambio de los medios de información para permitir establecer la sesión.
- La modificación de sesiones de comunicación existentes.
- Finalización de sesiones de comunicación existentes.

El SIP también se ha extendido a la petición y respuesta de información de presencia (condiciones de conectado o desconectado y localización de información como la contenida en una lista de amigos). Las funciones de presencia incluyen:

- Publicación y descarga de información de presencia.
- Petición y entrega de información de presencia.
- Presencia y otra notificación de evento.
- Acarreo de mensajes instantáneos.

El SIP se usa también para establecer un tipo entero de sesiones que no tienen ningún parecido con las llamadas telefónicas convencionales.

4.3. Línea de protocolos de Internet

La Figura 4.1 muestra las cuatro capas de la línea de protocolos multimedia de Internet. Las capas mostradas y los protocolos identificados serán descritos a continuación.

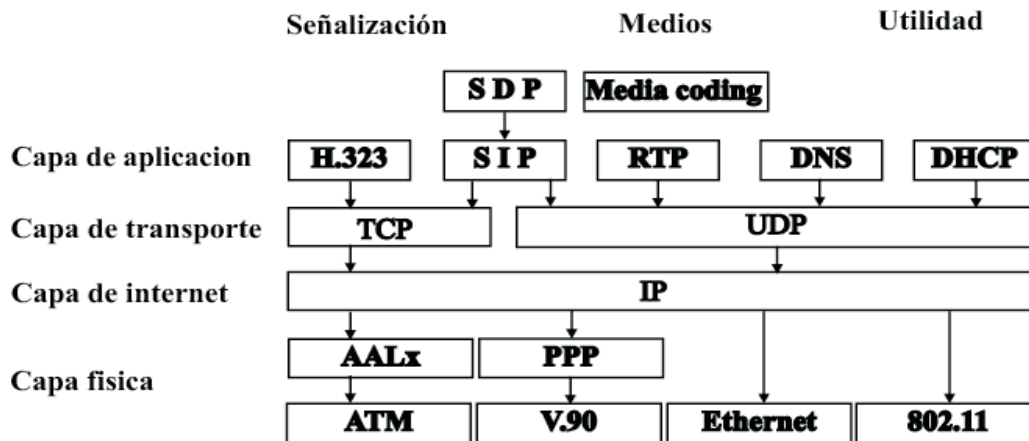


Figura 4.1: Línea de protocolos multimedia de Internet

4.3.1. Capa física

Las capas más bajas son la física y la capa de enlace. La capa física puede ser una Ethernet LAN, una línea telefónica (V.90 o MODEM DE 56k) funcionando en un protocolo PPP o una DSL funcionando en un modo de transporte ATM o incluso una red de trabajo inalámbrica 802.11. La capa de enlace realiza funciones tales como el intercambio del símbolo, sincronización de tramas y la especificación de la interfaz física.

4.3.2. Capa de Internet

La siguiente capa en la Figura 4.1 es la capa de Internet. En esta capa es utilizado el protocolo IP [16] para rutar un paquete a través de la red usando la dirección de destino IP. El protocolo IP es una interconexión, además de que es el mejor esfuerzo por desarrollar un protocolo de entrega de paquetes. Existen dos tipos de direcciones IP.

Las direcciones IPv4 tienen una longitud de cuatro octetos, normalmente escritas en la llamada notación "punto decimal" (por ejemplo, 207.134.3.5). Entre cada uno de los puntos está un número decimal entre 0 y 255. En la capa IP, los paquetes no pueden ser confirmados de que se han recibido. Un control total es calculado para detectar corrupción en la cabecera IP que podría causar que un paquete llegara a ser mal ruteado.

Sin embargo, los errores o corrupciones en la carga útil de la IP, no son detectados; para esto existe una capa más alta que debe realizar la función de detección de errores si es necesario. SIP usa un protocolo de números conformado por octetos simples en la cabecera del paquete para identificar el protocolo de la capa de transporte que recibirá el paquete.

La IPv6 [17] fue desarrollada por la IETF como un reemplazo para la IPv4. Ha ido ganando lentamente apoyo y ahora es soportada por un gran número de sistemas operativos. Inicialmente las redes más grandes de IPv6 probablemente fueron las portadoras de telefonía inalámbrica quienes necesitaban la ventaja más importante de IPv6 sobre IPv4 que es, un espacio de direccionamiento mucho mayor.

IPv6 incrementó el espacio de direccionamiento de 32 bits en el IPv4 a 128 bits, proporcionando arriba de 4 billones de direcciones IPv6. Hay muchas otras mejoras en IPv6 sobre IPv4, incluyendo la seguridad.

Una dirección de IPv6 es típicamente escrita como una secuencia de seis números hexadecimales separados por dos puntos. Por ejemplo, 0 : 0 : 0 : 0 : aaaa : bbbb : cccc : dddd, es una dirección IPv6 escrita en ese formato. También es común disminuir las secuencias de ceros con un simple doble dos puntos. La misma dirección puede entonces ser escrita como: aaaa : bbbb : cccc : dddd.

Las direcciones IP utilizadas en el Internet público son asignadas en bloques por la IANA. Como resultado de esta asignación centralizada, las direcciones IP son únicas globalmente. Estas habilitan un paquete para ser ruteado a través del Internet público utilizando solamente la dirección IP de destino.

4.3.3. Capa de transporte

La siguiente capa mostrada en la Figura 4.1 es la capa de transporte. Utiliza un número de puerto de dos octetos de la capa de aplicación para entregar el datagrama o segmento al protocolo de la capa de aplicación correcto en la dirección IP de destino.

Algunos números de puertos están dedicados exclusivamente a los protocolos de capa de aplicación, estos puertos son llamados números de puertos "well-known". Por ejemplo, HTTP utiliza el número de puerto well-known 80, mientras que SIP utiliza el número de puerto well-known 5060 o 5061.

Otros números de puertos pueden ser usados por otros protocolos y son asignados dinámicamente de un grupo de puertos disponibles. Estos llamados números de puertos efímeros están normalmente en el rango de 49152 a 65535. Hay tres protocolos comúnmente utilizados en la capa de transporte los cuales son: TCP, TLS y UDP.

TCP

TCP [18] proporciona, de manera confiable, conexión orientada a transporte sobre IP. Utiliza una secuencia de números y una confirmación de recepción positiva para asegurarse que cada bloque de datos, llamada, o segmento haya sido recibido. Los segmentos perdidos son retransmitidos hasta que sean exitosamente recibidos. La Figura 4.2 muestra el intercambio de mensajes para establecer y terminar una conexión TCP.

Un servidor TCP escucha de un puerto well-known una solicitud de TCP. El cliente TCP envía un mensaje SYN para abrir la conexión. El mensaje SYN contiene el número de secuencia inicial que el cliente usará durante la conexión. El servidor responde con un mensaje SYN conteniendo su propio número de secuencia inicial y un número de confirmación de recepción, indicando que fue recibido la SYN al cliente.

El cliente completa el protocolo de enlace tripartito con un ACK o un paquete de datos con la bandera AK colocada para el servidor con la confirmación de recepción del número de secuencia del servidor. Ahora que la conexión esta abierta, cualquiera de los dos, cliente o servidor, puede enviar datos en paquetes llamados segmentos.

Cada vez que un remitente transmite un segmento, se inicializa un cronómetro. Si

un segmento se pierde en la transmisión, este cronómetro expirará. Deduciendo un segmento perdido, el remitente reenvía el segmento hasta que se recibe la confirmación de recepción. El mensaje FIN cierra la conexión TCP. La secuencia de cuatro mensajes mostrada en la Figura 4.2 cierra la conexión.

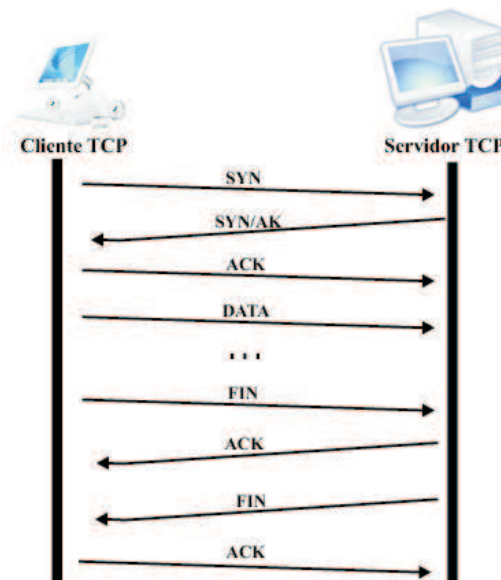


Figura 4.2: Abriendo y cerrando una sesión SIP

UDP

UDP [19] proporciona un transporte inestable a través del Internet. Aunque es el mejor esfuerzo para la entrega de servicios desde que no hay una confirmación de recepción de datagramas enviados. No presenta mucha de la complejidad de TCP incluyendo números de secuencia, confirmaciones de recepción y tamaños de ventanas.

UDP no detecta errores en los datagramas con un control completo. Depende de los protocolos de las capas más altas para detectar los datagramas perdidos e iniciar una retransmisión si es necesario.

TLS

TLS [20] está basado en el protocolo de seguridad en capas de conexión SSL utilizado primero en navegadores Web y utiliza TCP para transporte. TLS es comúnmente usado ahora en el Internet para la seguridad de sitios Web utilizando el esquema de URI. El protocolo TLS tiene a su vez dos capas: el protocolo de transporte TLS y el protocolo de enlace TLS.

El protocolo de transporte TLS es usado para proporcionar un mecanismo de transporte confiable y privado. El dato enviado utilizando el protocolo de transporte TLS es encriptado para que terceros no puedan interceptarlo. Un tercero tampoco puede modificar el dato transportado sin que uno de los participantes descubra esto.

El protocolo de enlace TLS es usado para establecer la conexión, gestiona las llaves de encriptación utilizadas por el protocolo de transporte TLS y proporciona autenticación. El esquema de gestión de llaves selecciona un algoritmo de encriptación y genera una llave basada en una comunicación secreta entre los dos lados. Durante el enlace, los participantes intercambian certificados, los cuales pueden ser usados para la autenticación.

Muchos sistemas operativos ya soportan TLS debido a que es ampliamente utilizado en la seguridad de servidores y navegadores Web. La Figura 4.3 muestra como se ejecuta la conexión con TLS.

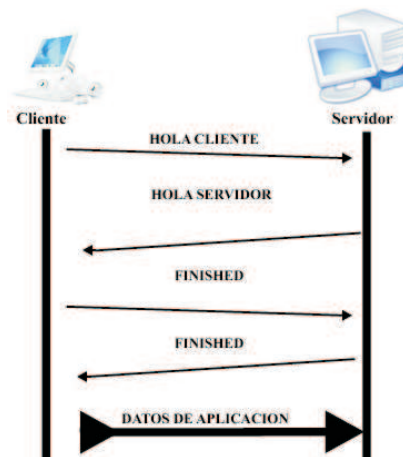


Figura 4.3: Abriendo conexiones TLS

SCTP

El protocolo SCTP [21] es similar al protocolo TCP en que proporciona un transporte confiable basado en la secuencia del flujo de datos. Sin embargo, tiene algunas ventajas sobre el transporte que ofrece TCP para un protocolo basado en mensajes.

Primero, incorpora la segmentación de mensajes, de tal forma que los mensajes individuales son separados en la capa de transporte. Otra ventaja es que SCTP evita la supuesta cabeza de bloque de línea que es un problema de TCP.

SCTP también soporta multimensajería, así que si uno de un par de servidores de descarga equilibrada falla, el otro puede inmediatamente comenzar a recibir los mensajes incluso sin necesitar un DNS o de otra base de datos.

SCTP es un verdadero protocolo de la capa de transporte que necesita el soporte de un sistema operativo nivelado para ser usado y que inicialmente su uso esta retrasado en el Internet. También es importante notar que las ventajas de SCTP sobre TCP solamente se presentan durante la pérdida de paquetes. En una red con cero pérdidas, el rendimiento de los dos es idéntico.

4.3.4. Capa de aplicación

La capa más alta que se muestra en la Figura 4.1 es la capa de aplicación. Esta incluye protocolos de señalización como lo es SIP, protocolos de transporte de medios de comunicación como RTP y se incluye también a H.323, el cual es un protocolo de señalización alternativo para SIP desarrollado por la ITU.

El protocolo SDP, se muestra encima de SIP en la columna de protocolos porque es transportado en un cuerpo de mensaje de SIP. HTTP, SMTP, FTP y Telnet son todos los ejemplos de protocolos en la capa de aplicación. Debido a que SIP puede utilizar cualquier protocolo de transporte, en la Figura 4.1 se muestra interactuando ya sea con TCP, TLS y UDP.

4.4. Métodos de solicitud

Los métodos de solicitud en el SIP son considerados los "verbos" del protocolo, desde que se realiza la solicitud principal o una acción en específico que será conducida por otro UA o servidor. Los seis métodos principales en SIP se definen por completo en la RFC 3261 [13] y son:

- INVITE
- REGISTER
- BYE
- ACK
- CANCEL
- OPTIONS

De estos seis métodos REGISTER, INVITE, ACK, CANCEL son utilizados para el proceso de inicio de sesiones, BYE es utilizado para terminar estas sesiones y OPTIONS para describir y especificar las capacidades del servidor.

- INVITE (Invitación)

El método INVITE es utilizado para establecer sesiones multimedia entre agentes de usuarios. En telefonía, esto es similar a un SETUP Messages (Estructuración de mensajes) con ISDN. Un INVITE regularmente tiene un cuerpo de mensaje que contiene la información acerca de los medios de comunicación del participante a la sesión. El cuerpo del mensaje puede también contener otra información acerca de la sesión como la QoS o información de seguridad.

Si un INVITE no contiene información de los medios de comunicación, el método ACK contendrá la información de los medios de comunicación del UAC, un ejemplo de esto puede observarse en la Figura 4.4.

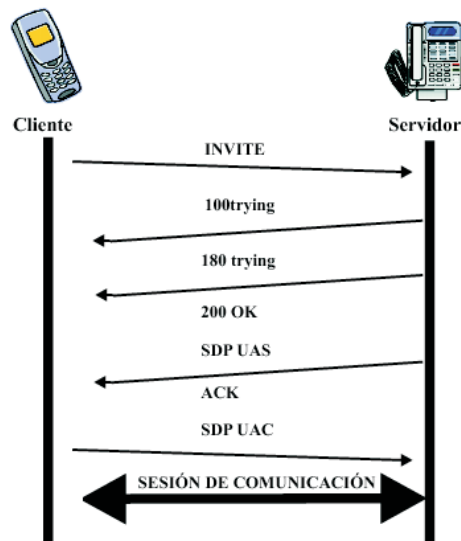


Figura 4.4: INVITE con cuerpo de mensaje

Si la información de los medios de comunicación contenida en el ACK no es aceptable entonces el participante a la sesión o invitado deberá enviar un método BYE para cancelar la sesión, esto debido a que no puede enviarse un CANCEL por que la sesión no ha sido todavía establecida.

Se considera establecida una sesión de medios de comunicación cuando en el método INVITE se presenta un código de estado 200 OK y el mensaje ACK a sido intercambiado entre el agente UAC y el agente UAS.

Una solicitud de INVITE exitoso puede establecer un diálogo entre los dos agentes de usuarios, esta sesión continuará hasta que un método BYE sea enviado por cualquier participante para terminar la sesión.

Un agente UAC que origina un INVITE para establecer un diálogo o conversación, crea un Call-ID único y global que será utilizado durante la duración de la llamada. La combinación de la etiqueta TO, la Etiqueta FROM y el Call-Id es la única identificación para el diálogo establecido.

Un contador CSeq se inicializa y se incrementará por cada nueva solicitud para el mismo Call-ID, Las cabeceras TO y FROM son llenadas con la dirección remota y la dirección local respectivamente. Una etiqueta FROM es incluida en la INVITE, y el UAS incluye

una etiqueta TO en toda respuesta.

Se conoce como re-INVITE cuando se envía un INVITE por un diálogo existente y se hace referencia al mismo Call-Id que el original INVITE y contiene también las mismas etiquetas TO y FROM . Esta solicitud es utilizada para cambiar las características de la sesión o para actualizar del diálogo. El numero de la secuencia de comando Cseq se irá incrementando de tal forma que un UAS puede distinguir entre un re-INVITE y una retransmisión del INVITE original.

Una cabecera EXPIRES en un INVITE indica al UAS la duración de validez para la solicitud de llamada. Una vez que es establecida la sesión, la cabecera EXPIRES no tiene sentido, la expiración de este tiempo no terminara la sesión multimedia. En cambio, una cabecera SESSION-EXPIRES puede ser usada para poner un tiempo límite para un establecimiento de sesión. Un ejemplo de solicitud INVITE con cuerpo de mensaje SDP se muestra en la Tabla 4.1.

```
INVITE sip:411@salzburg.at:user=phone SIP/2.0
Via: SIP/2.0/UDP salzburg.edu.at:5060;branch=z9hG4bK1d32hr4
Max-Forwards:70
To: <sip:411@salzburg.at:user=phone>
From: Christian Doppler <sip:c.doppler@salzburg.edu.at>
;tag=817234
Call-ID: 12-45-A5-46-F5@salZbUrg.edu.at
CSeq: 1 INVITE
Subject: Train Timetables
Contacto sip:c.doppler@salzburg.edu.at
Content-Type: application/sdp
Content-Length: 151
v=0
o=doppler 2890842326 2890844532 IN IP4 salzburg.edu.at
s=Phone Call
c=IN IP4 50.61.72.83
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000
```

Tabla 4.1: Ejemplo de solicitud INVITE

Como extra a las cabeceras requeridas, la solicitud que se mostró como ejemplo contiene opcionalmente la cabecera SUBJECT (asunto). Las cabeceras de campo obligatorias en una solicitud INVITE se muestran en la Tabla 4.2

- REGISTER (Registro)

El método REGISTER es utilizado por un agente de usuario para notificar a una red de trabajo SIP su recurso actual de identificación uniforme de contacto CONTACT-URI

Call-ID
Cseq
To
Via
Contact
Max-forwards

Tabla 4.2: Cabeceras de campo obligatorio en una solicitud INVITE

(dirección IP) y la URI que deberá tener para rutar la solicitud a este CONTACT.

Un SIP REGISTER lleva algunas similitudes con un registro inicial en telefonía celular. Una solicitud de REGISTER puede contener un cuerpo de mensaje, aunque su uso no está definido en los estándares.

Dependiendo del uso de las cabeceras CONTACT y EXPIRES en la solicitud REGISTER, el servidor de registro puede tomar diferentes acciones. Si no existen parámetros de vencimiento o la cabecera EXPIRE está presente, un SIP URI terminará su vigencia en 1 hora. La presencia de una cabecera EXPIRES establece también el vencimiento, duración o vigencia para contactos que no contengan parámetros de vencimiento.

Si existen parámetros de expiración presentes, entonces dichos parámetros establecerán la vigencia solamente para el contacto que los contenga. Las URIs que no corresponden al protocolo SIP no tienen un tiempo de vencimiento predefinido.

Un agente de usuario que envía una solicitud de REGISTER puede recibir una respuesta con código de estado 3xx redirección o 4xx error conteniendo las cabeceras de contacto de la localización a la cual debe ser enviado. Un ejemplo de una solicitud de registro para el usuario EUCLID se muestra en la Tabla 4.3.

Las cabeceras obligatorias en una solicitud REGISTER (registro) se muestran en la Tabla 4.4

- BYE (Adiós)

El método BYE se utiliza para terminar una sesión multimedia establecida. En telefonía, es similar a un mensaje de liberación. Una sesión se considera establecida si una INVITE

```

REGISTER sip:registrar.athens.gr SIP/2.0
Via: SIP/2.0/UDP 201.202.203.204:5060;branch=z9hG4bK313
Max-Forwards:70
To: sip:euclid@athens.gr
From: <sip:secretary@academy.athens.gr>;tag=543131
Call-ID: 2000-July-O7-23:59:59.1234@201.202.203.204
CSeq: 1 REGISTER
Contact: sip:euclid@parthenon.athens.gr
Contact: mailto:euclid@geometry.org
Content-Length: 0

```

Tabla 4.3: Ejemplo de solicitud de registro para el usuario EUCLID

Call- ID
Cseq
From
To
Via
Max-forwards

Tabla 4.4: Cabeceras de campo obligatorias en una solicitud REGISTER

a recibido una respuesta exitosa de clase 2xx o un ACK ha sido enviado.

Un BYE es enviado solamente por un UA participante en la sesión, nunca por proxies u otros terceros implicados. Es un método end-to-end, así que las respuestas son solamente generadas por el otro UA. Un agente de usuario UA responde con un 481 Dialog /Transaction Does Not Exist (diálogo/transacción no existente) a un BYE para un diálogo desconocido que no corresponda a la sesión establecida. Un ejemplo de una solicitud BYE se puede apreciar en la Tabla 4.5.

Las cabeceras obligatorias para una solicitud BYE (adiós) se muestran en la Tabla 4.6

- ACK (Confirmación de recepción)

El método ACK es utilizado para confirmar la recepción final a una solicitud INVITE. Las respuestas finales a muchas otras solicitudes no son reconocidas con la confirmación de recepción. Las respuestas finales son las definidas con las contestaciones de clase 2xx,

```

BYE sip:info@hypotenuse.org SIP/2.0
Via: SIP/2.0/UDP port443.hotmail.com:5060;branch=z9hG4bK312bc
Max-Forwards:70
To: <sip:info@hypotenuse.org>;tag=63104
From: <sip:pythag42@hotmail.com>;tag=9341123
Call-ID: 34283291273@port443.hotmail.com
CSeq: 47 BYE
Content-Length:0

```

Tabla 4.5: Ejemplo de solicitud BYE

Call- ID
Cseq
From
To
Via
Max-forwards

Tabla 4.6: Cabeceras de campo obligatorio de una solicitud BYE

3xx, 4xx, 5xx o 6xx.

Un ACK puede contener un cuerpo de mensaje SDP como aplicación (APPLICATION/SDP). Esto es permitido si la INVITE inicial no contiene en sí un cuerpo de mensaje SDP. Si la INVITE contiene un cuerpo de mensaje de este tipo, entonces el ACK no debe contener este tipo de cuerpo de mensaje.

La ACK no puede ser utilizado para modificar la descripción de una sesión multimedia que ha sido ya enviada en la INVITE inicial; en todo caso para tal propósito deber ser usado la solicitud re-INVITE.

El protocolo SDP es utilizado en un ACK solo para algunos escenarios de trabajo interno con otros protocolos en donde las características multimedia no pueden conocer en que momento la INVITE original ha sido generada y enviada. Las cabeceras obligatorias en una confirmación de recepción ACK se muestran en la Tabla 4.7

- CANCEL (Cancelar)

Call- ID
Cseq
From
To
Via
Max-forwards

Tabla 4.7: Cabeceras de campo obligatorio en una solicitud ACK

El método CANCEL es usado para terminar búsquedas en proceso o intentos de llamadas. Puede ser generado por cualquier UA permitiendo que una respuesta de clase 1xx sea recibida, pero una respuesta final no se ha recibido. Un UA utiliza el método para cancelar un intento de llamada en curso.

La identificación filial o de sucursal BRANCH ID para el CANCEL debe coincidir con la INVITE que se esta cancelando. Un CANCEL solo tiene significado para una INVITE puesto que solamente una INVITE puede tomarse unos segundos (o minutos) para completarla.

Todas las otras solicitudes en SIP la completan inmediatamente. Por consiguiente, el resultado final será siempre generado antes de que el CANCEL es recibido. Las cabeceras obligatorias en una solicitud CANCEL (cancelar) se muestran en la Tabla 4.8

Call- ID
Cseq
From
To
Via
Max-forwards

Tabla 4.8: Cabeceras de campo obligatorio en una solicitud CANCEL

- OPTIONS (Opciones)

El método OPTIONS es utilizado para consultar a un agente de usuario o servidor acerca de sus capacidades o características y descubrir su disponibilidad actual. La respuesta a esta solicitud será una lista con las características del agente de usuario UA o del servidor.

Un agente de usuario UA o servidor responde a esta solicitud como si fuera una INVITE por ejemplo, si no esta aceptando llamadas, contestará con una respuesta de clase 4xx o 6xx. Las cabeceras obligatorias en una solicitud OPTIONS (opciones) son las mismas que en la Tabla 4.4

- REFER (Referencia)

El método REFER [22] es usado por un UA para solicitarle a otro agente de usuario el acceso a la URI o a la URL. El recurso es identificado por una URI o URL en la cabecera requerida de campo REFER-TO (referencia para).

Cabe notar que tanto la URI como la URL pueden ser de cualquier tipo de recurso de identificación o de localización, esto es: SIP, SIPs, HTTP y mas. Una solicitud de REFER puede ser enviada ya sea dentro o fuera de un diálogo existente.

Un flujo típico de llamada para esta solicitud se muestra en la Figura 4.5. En esté ejemplo, un UAC envía una solicitud de REFER a un UAS. El UAS, después de llevar a cabo cualquier autorización e identificación, decide aceptar la solicitud de REFER y responde con una contestación 202 ACCEPTED.

Es importante observar que esta respuesta es enviada inmediatamente sin esperar por que la activación de la solicitud este completa.

Un ejemplo de un mensaje REFER (referencia) se muestra en la Tabla 4.9.

La Tabla 4.10 lista las cabeceras de campo obligatorias en una solicitud de REFER (referencia).

- SUBSCRIBE (Subscribir)

El método SUBSCRIBE [23] es utilizado por un UA para establecer una suscripción con el propósito de recibir notificaciones (vía el método NOTIFY) acerca de un evento en

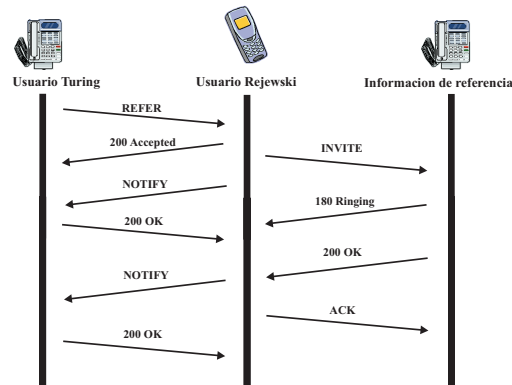


Figura 4.5: Flujo de llamada para una REFER (referencia)

```

REFER sip:m.rejewski@biuroszyfrow.p1 SIP/2.0
via SIP/2.0/UDP 1ab34.b1etch1eypark.co.uk:5060
;branch=z9hG4bK932039
Max-Forwards: 69
To: <sip:m.rejewski@biuroszyfrow.p1>;tag=ACEBDC
From: Alan Turing <sip:turing@b1etch1eypark.co.uk>;tag=213424
Call-ID: 3419fak3kFD23s1A9dk1
CSeq: 5412 REFER
Refer-To: <sip:info@scherbius-ritter.com>
Content-Length: 0
  
```

Tabla 4.9: Ejemplo de un mensaje de solicitud REFER

particular. Una suscripción exitosa establece un diálogo entre el agente UAC y el agente UAS.

La solicitud de suscripción contiene una cabecera de campo EXPIRES, la cual indica la duración deseada para la existencia de la suscripción. Después de que este periodo de tiempo pasa, la suscripción es terminada automáticamente.

Una respuesta 202 ACCEPTED (aceptado) a un SUBSCRIBE no indica si la suscripción ha sido autorizada, simplemente significa que ha sido entendida por el servidor.

El flujo de llamada básico para este método se muestra en la Figura 4.6. El cliente envía una solicitud de SUBSCRIBE, la cual es exitosa, y recibe NOTIFYs (notificaciones). Antes del vencimiento del tiempo de suscripción, el cliente re-SUBSCRIBE (resuscribe) para extender la suscripción y recibir mas notificaciones.

To
From
Call-ID
Cseq
Contact
Max-forwards
Via
Refer-to

Tabla 4.10: Cabeceras de campo obligatorio para una solicitud REFER

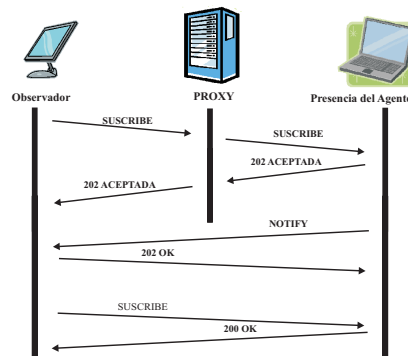


Figura 4.6: Llamada de un SUSCRIBE y NOTIFY

Un ejemplo de una solicitud de SUBSCRIBE (suscribir) se muestra en la Tabla 4.11.

El tipo de evento de suscripción es indicado por la cabecera de campo EVENT (evento) requerida en la solicitud de SUBSCRIBE. Cada aplicación de la estructura de eventos en SIP [23] define un paquete con una única etiqueta de EVENT. Cada paquete define las siguientes cosas:

1. El intervalo de expiración o vencimiento de suscripción predefinido.
2. Cuerpos de mensaje esperados de SUBSCRIBE.
3. Que eventos causan que una NOTIFY sea enviada y que cuerpos de mensaje son esperados en la NOTIFY.

```

SUBSCRIBE sip:ptolemy@rosettastone.org SIP/2.0
Via SIP/2.0/UDP proxy.elasticity.co.uk:5060
;branch=z9hG4bK348471123
Via SIP/2.0/UDP parlour.elasticity.co.uk:5060
;branch=z9hG4bKABDA ;received=192.0.3.4
Max-Forwards: 69
To: <sip:Ptolemy@rosettastone.org>
From: Thomas Young <sip:tyoung@elasticity.co.uk>;tag=1814
Call-ID: 452k59252058dkfj349241k34
CSeq: 3412 SUBSCRIBE
Allow-Events: dialog
Contact: <sip:tyoung@parlour.elasticity.co.uk>
Event: dialog
Content-Length: 0

```

Tabla 4.11: Ejemplo de una solicitud de SUSCRIBE

4. Si la NOTIFY contiene sus condiciones completas o incrementarlas (detalles).
5. Proporción máxima de notificaciones.

La Tabla 4.12 lista las características actuales de eventos SIP y paquetes de plantillas.

Conference	Conferencia de información incluyendo lista de participantes.
Dialog	Estado de dialogo e información de identificación.
Message-summary	Notificación de mensajes.
Presence	Presencia de información.
Refer	Referencia.
Reg	Estado del registro de usuario.
Winfo	Información del observador.

Tabla 4.12: Paquetes de eventos y paquetes de plantillas

La Tabla 4.13 muestra las cabeceras de campo obligatorias en una solicitud SUBSCRIBE (suscribir) [23].

To	Via
From	Contact
Call-ID	Event
Cseq	Allow-events
Max-forwards	

Tabla 4.13: Cabeceras de campo obligatorio para una solicitud SUSCRIBE

- NOTIFY (Notificación)

El método NOTIFY [23] es utilizado por agente de usuario UA para transportar información acerca de la ocurrencia de un evento en particular. Un NOTIFY siempre es enviado conteniendo un diálogo cuando una suscripción existe entre un suscriptor y un notificador. Un flujo de llamada básico mostrando el método NOTIFY se muestra en la Figura 4.6.

Una solicitud de NOTIFY recibe normalmente una respuesta 200 OK indicando que ha sido recibida. Si se recibe una respuesta 481 Dialog/Transaction does not exist (transacción de diálogo no existe), la suscripción es automáticamente terminada y no son enviadas mas NOTIFYs.

Una solicitud de NOTIFY contiene una cabecera de campo EVENT (evento) indicando el paquete y una cabecera de campo SUBSCRIPTION-STATE (estado de la suscripción) indicando el estado actual de la suscripción. La cabecera de campo EVENT contendrá el nombre del paquete usado en la suscripción.

Una NOTIFY siempre es enviada en la principio de una suscripción y también en la terminación de una suscripción. Un ejemplo de una solicitud de NOTIFY (notificación) se muestra en la Tabla 4.14.

```
NOTIFY sip:tyoung@parlour.elasticityoco.uk SIP/200
via SIP/2.0/UDP cartoucheorostastone.org:5060
;branch=z9hG4bK3841323
Max-Forwards: 70
To: Thomas Young <sip:tyoung@elasticityoco.uk>;tag=1814
From: <sip:ptolerny@rosettastone.org>;tag=5363956k
Call-ID: 452k59252058dkfj349241k34
CSeq: 3 NOTIFY
Contact: <sip:Ptolerny@cartouche.rosettastone.org>
Event: dialog
Subscription-State: active
Allow-Events: dialog
Content-Type: application/xml+dialog
Content-Length: ...
(XML Message body not shown. 00)
```

Tabla 4.14: Ejemplo de una solicitud NOTIFY

La Tabla 4.15 lista las cabeceras de campo obligatorias en una solicitud de NOTIFY (notificación).

- MESSAGE (Mensaje)

To
From
Call-ID
Cseq
Max-forwards
Via
Contact
Event
Suspension-State
Allow-Events

Tabla 4.15: Cabeceras de campo obligatorio en una solicitud NOTIFY

El método MESSAGE [24] es utilizado para transportar mensajes instantáneos (IM) utilizando SIP. Un IM normalmente consiste de un mensaje corto intercambiado casi en tiempo real por los participantes inmersos en una conversación. Los MESSAGEs pueden ser enviados conteniendo un diálogo o con un diálogo externo, pero no pueden establecer un diálogo por si mismos.

Los mensajes actuales contenidos son transportados en el cuerpo del mensaje como una conexión MIME. Todos los UAs que soportan el método MESSAGE deben soportar el formato PLAIN/TEXT, también deben soportar otros formatos como MESSAGE/CPIM [25] o TEXT/HTML o muchos otros.

Una solicitud de MESSAGE normalmente recibe una respuesta 200 OK indicando que el mensaje ha sido entregado a su destinación final. Una respuesta a un IM no debe ser enviada en el cuerpo del mensaje de una respuesta 200 OK, si no mas bien en una solicitud de MESSAGE separada enviada al remitente original. Un ejemplo de flujo de llamada de MESSAGE se muestra en la Figura 4.7.

Una respuesta 202 ACCEPTED (aceptada) indica que la solicitud ha alcanzado un dispositivo STORE-AND-FORWARD (almacenar y remitir) y finalmente en el futuro será entregada a su destinación final. En ningún caso la respuesta 200 OK confirma que el mensaje haya sido devuelto al usuario.

Un ejemplo de solicitud de MESSAGE se muestra en la Tabla 4.16.

La Tabla 4.17 lista las cabeceras de campo obligatorias en una solicitud de MESSAGE.

- INFO (Información)

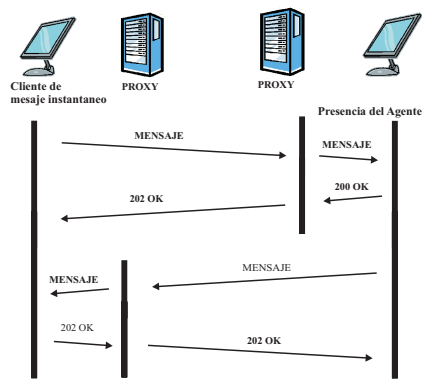


Figura 4.7: Mensajería instantánea SIP

```
MESSAGE sip:editor@rcs.org SIP/2.0
via SIP/2.0/UDP lab.mendelev.org:5060;branch=z9hG4bK3
Max-Forwards: 70
To: <editor@rcs.org>
From: "D. I. Mendelev" <dmitry@mendelev.org>;tag=1865
Call-ID: 93847197172049343
CSeq: 5634 MESSAGE
Subject: First Row
Contact: <sip:dmitry@lab.mendelev.org>
Content-Type: text/plain
Content-Length: 5
```

Tabla 4.16: Ejemplo de una solicitud de MESSAGE

El método INFO [26] es usado por un agente de usuario UA para enviar una señalización de llamada informando a otro UA con el cual ha establecido una sesión multimedia. Esto es diferente de un re-INVITE (reinvitación) puesto que este método no cambia las características multimedia de la llamada.

Para diálogos desconocidos la solicitud de INFO (información) recibirá una respuesta 481 Transaction/Dialog does not exist (transacción de diálogo no existe). Este método contiene típicamente un cuerpo de mensaje. El contenido puede ser información de señalización, eventos de llamada, o alguna clase de incentivos.

INFO ha sido propuesto para transportar cierta información de señalización de llamadas PSTN tal como mensajes ISUP, USR. El método INFO incrementa siempre al número Cseq. Un ejemplo del método INFO (información) se muestra en la Tabla 4.18.

Las cabeceras de campo obligatorias en una solicitud INFO (información) son mostradas en la Tabla 4.19

To
From
Call-ID
Cseq
Max-Forwards
Via

Tabla 4.17: Cabeceras de campo obligatorio en una solicitud MESSAGE

```

INFO sip:poynting@mason.edu.uk SIP/2.0
via: SIP/2.0/UDP cavendish.kings.cambridge.edu.uk;
branch=z9hG4bK24555
Max-Forwards: 70
To: John Poynting <sip:nting@mason.edu.uk> ;tag=3432
From: J.C. Maxwell <sip:james.maxwell@kings.cambridge.edu.uk>
;tag=432485820183
Call-ID: 18437@cavendish.kings.cambridge.edu.uk
CSeq: 6 INFO
Content-Type: message/isup
Content-Length: 16
51a6324134527

```

Tabla 4.18: Ejemplo de solicitud INFO

- PRACK (Confirmación de recepción provisional)

El método PRACK es utilizado para confirmar la recepción de respuestas provisionales transportadas fiablemente (respuestas de clase 1xx). La confiabilidad de respuestas de clases 2xx, 3xx, 4xx, 5xx y 6xx para solicitudes de INVITEs es alcanzada utilizando el método ACK. Sin embargo, en casos donde una respuesta provisional es recibida, como 180 Ringing (sonando), es crítico determinar el estado de la llamada, además de que es necesario para el receptor de una respuesta provisional que le sea confirmada.

El método PRACK se aplica para toda respuesta provisional excepto la respuesta 100 Trying, la cual nunca es confiable transportar. Una PRACK es generada por un UAC cuando ha sido recibida una respuesta provisional conteniendo un RSEQ número de secuencia confiable, y una cabecera SUPPORTED (soporte)

La llamada se completa cuando el agente de usuario cliente UAC envía la ACK en respuesta a la 200 OK. Como se muestra en la Figura 4.8, la PRACK recibe una respuesta 200 OK, la cual puede ser distinguida de la respuesta 200 OK para la INVITE por el método contenido en la cabecera Cseq.

Mandatory headers
Call-ID
Cseq
From
To
Via
Max-Forwards

Tabla 4.19: Cabeceras de campo obligatorio en una solicitud INFO

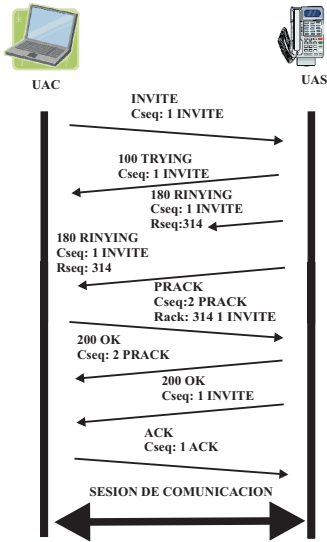


Figura 4.8: Uso de respuestas provisionales confiables

El método PRACK siempre incrementa al número Cseq. Una PRACK puede contener un cuerpo de mensaje. Un ejemplo de intercambio de respuestas provisionales se muestra en la Tabla 4.20.

Las cabeceras de campo obligatorias en una solicitud de PRACK se muestra en la Tabla 4.21.

■ UPDATE (Actualizar)

El método UPDATE es utilizado para modificar el estado de una sesión sin cambiar el estado del diálogo. En SIP una sesión es establecida utilizando una solicitud INVITE en un modo de propuesta de contestación. Típicamente, una propuesta de sesión es hecha en la INVITE y una contestación es hecha en una respuesta para la INVITE.

```

SIP/2.0 180 Ringing
via: SIP/2.0/UDP lucasian.trinity.cambridge.edu.uk
;branch=z9hG4bK452352
;received=1.2.3.4
To: Descartes <sip:rene.descartes@metaphysics.org>;tag=12323
From: Newton <sip:newton@kings.cambridge.edu.uk>;tag=981
Ca11-ID: 5@lucasian.trinity.cambridge.edu.uk
RSeq: 314
CSeq: 1 INVITE
Content-Length: 0

PRACK sip:rene.descartes@metaphysics.org SIP/2.0
Via: SIP/2.0/UDP lucasian.trinity.cambridge.edu.uk
;branch=z9hG4bKdtyw
Max-Forwards: 70
To: Descartes <sip:rene.descartes@metaphysics.org>;tag=12323
From: Newton <sip:newton@kings.cambridge.edu.uk>;tag=981 Ca11-ID: 5@lucasian.trinity.cambridge.edu.uk
CSeq: 2 PRACK
RAck: 314 1 INVITE
Content-Length: 0

SIP/2.0 200 OK
Via: SIP/2.0/UDP lucasian.trinity.cambridge.edu.uk
;branch=z9hG4bKdtyw ;received=1.2.3.4
To: Descartes <sip:rene.descartes@metaphysics.org>;tag=12323
From: Newton <sip:newton@kings.cambridge.edu.uk>;tag=981 Ca11-ID: 5@lucasian.trinity.cambridge.edu.uk
CSeq: 2 PRACK
Content-Length: 0

```

Tabla 4.20: Ejemplo de intercambio de respuestas provisionales

En un establecimiento de sesión, una re-INVITE se utiliza para actualizar los parámetros de la sesión. Sin embargo, ningún participante en una sesión pendiente (invitación enviada pero sin respuesta final recibida) puede realizar esta petición, en su lugar, se utiliza el método UPDATE, en la Tabla 4.22 se muestra un ejemplo de un mensaje UPDATE. Los usos posibles de UPDATE incluye silenciamiento o colocación en espera del flujo de medios de comunicación pendientes, realizando QoS u otro atributo de negociación posterior al establecimiento de la sesión.

Las cabeceras de campo obligatorias para una solicitud UPDATE se muestran en la Tabla 4.23.

- TAGS (Etiquetas)

Una TAG, criptográficamente, es un número aleatorio con por lo menos 32 bits de aleatoriedad, la cual es adicionada a las cabeceras TO y FROM únicamente para identificar un diálogo.

La cabecera TO en la INVITE inicial no contendrá una etiqueta. Un visitante deberá incluir una etiqueta en la cabecera FROM, aunque un UA generalmente no hará lo que

Mandatory headers
Call-ID
Cseq
From
To
Via
Max-Forwards
Rack

Tabla 4.21: Cabeceras de campo obligatorio en una solicitud PRACK

```

UPDATE sips:beale@bufords.bedford.va.us SIP/2.0
Via SIP/2.0/TLS client.crypto.org:5060;branch=z9hG4bK342
Max-forwards: 70
To:t. Beale<sips:beale@bufords.bedford.va.us>;tag=71
From: Blaise Vigenere <sips:bvigenere@crypto.org>;tag=19438
Call-ID 170189761183162948
Cseq:94 UPDATE
Contact: <sips:client.crypto.org>
Content-Type: applications/sdp
Content-Length

```

Tabla 4.22: Ejemplo de un mensaje UPDATE

era opcional en la especificación RFC 2543. Excluyendo a la respuesta 100 TRYING todas las demás tendrán una etiqueta adicionada a la cabecera TO.

- MESSAGE BODIES (Cuerpos de mensaje)

Los MESSAGE BODIES en SIP contienen varios tipos de información. Pueden tener información SDP, la cual se utilizaba para transportar información multimedia, QoS ó inclusive información de seguridad.

La cabecera opcional CONTENT- DISPOSITION (disposición de contenido) es utilizada para indicar la intención de uso del cuerpo de mensaje. Si no esta presente, la función es asumida como una sesión, lo cual significa que el cuerpo describe una sesión multimedia. El formato del cuerpo de mensaje es indicado por la cabecera CONTENT-TYPE (tipo de contenido).

El esquema de codificación del cuerpo de mensaje es indicado en la cabecera CONTENT-ENCODING (codificación de contenido). Si no esta especificado, un esquema de codi-

Call-ID
Cseq
From
To
Via
Max-Forwards
Contact

Tabla 4.23: Cabeceras de campo obligatorio en una solicitud UPDATE

ficación se asume predefinidamente siendo este TEXT/PLAIN. Las especificaciones del esquema CONTENT-ENCODING le permite al cuerpo del mensaje comprimirse.

La cabecera CONTENT-LENGTH (extensión del contenido) contiene el número de octetos en el cuerpo del mensaje. Debido a que pueden ser enviados múltiples mensajes SIP en un flujo TCP, la cabecera CONTENT-LENGTH es un camino confiable para detectar cuando un mensaje termina y otro más inicia.

Si no está presente la cabecera CONTENT-LENGTH, el UAC debe asumir que el cuerpo del mensaje continua hasta el final del datagrama UDP, o hasta que la conexión TCP este cerrada, dependiendo del protocolo de transporte.

4.5. Tipos de respuesta

Hay seis clases de respuestas de SIP. Las primeras cinco clases se tomaron prestadas de HTTP; la sexta fue creada por SIP. Las respuestas descritas se definen por completo en la RFC 3261 [13]. Las clases de respuestas en SIP se muestran en el Cuadro 4.24.

Si un código de respuesta en particular de SIP no es entendida por el UAC, debe interpretarse por la clase de la respuesta. Por ejemplo, una respuesta desconocida 599 SERVER UNPLUGGED (servidor desconectado) debe ser interpretada por un agente de usuario como una respuesta de clase 500 SERVER FAILURE (falla del servidor).

Las expresiones razonadas son solamente para concepción humana, el protocolo SIP utiliza solo el código de respuesta en determinados comportamientos. Así que, una respuesta 200 CALL FAILED (fallo llamada) es interpretada de la misma forma que una respuesta 200 OK. Las expresiones razonadas escuchadas aquí son las sugeridas

Clase	Descripción	Acción
1xx	Informational	Indica el estado de la llamada
2xx	Success	Respuesta del suceso
3xx	Redirection	Servidor ha encontrado posible localización
4xx	Client error	Respuesta del cliente fallida
5xx	Server failure	Respuesta del servidor fallida
6xx	Global failure	Respuesta global fallida

Tabla 4.24: Clases de respuesta SIP

por el documento RFC.

Los nuevos códigos de respuesta creados por SIP empiezan típicamente x80 para tratar de evitar colisiones con los códigos de respuesta de HTTP. A menos que haya otro tipo de referencias, las respuestas descritas aquí están definidas en la RFC 3261.

Clase 1XX INFORMATIONAL (Informativo)

Las respuestas de clase informativa 1xx son usadas para indicar el progreso de la llamada. Las respuestas informativas son respuestas end-to-end y pueden contener cuerpos de mensaje. La excepción a estas es la respuesta 100 TRYING (intentando), la cual es solamente una respuesta hop-by-hop y puede no contener un cuerpo de mensaje.

Pueden ser enviadas cualquier número de respuestas informativas por un UAS antes de que una respuesta final (respuestas de clase 2xx, 3xx, 4xx, 5xx o 6xx) sea enviada.

La primera respuesta informativa recibida por el UAC confirma la recepción de la INVITE y detiene la retransmisión de la INVITE. Por esta razón, los servidores regresan la respuesta 100 TRYING minimizando la retransmisión de la INVITE en la red de trabajo. Además, las respuestas informativas no tienen efecto en las retransmisiones de la INVITE.

Todas las respuestas provisionales con excepción de 100 TRYING deben contener una CONTACT URI (recurso de identificación uniforme de contacto) y repetir todas las cabeceras RECORD-ROUTE (registro de ruta) recibidas en la solicitud.

- 100 Trying (Intentando)

La respuesta 100 Trying es solo una solicitud hop-by-hop. Nunca es remitida y puede no contener un cuerpo de mensaje. La respuesta 100 Trying puede ser generada ya sea por un servidor Proxy o un UA. Solamente indica que algunos tipos de acciones están siendo tomadas para procesar la llamada, no indica que el usuario ha sido localizado. Una respuesta 100 TRYING típicamente no contiene una etiqueta TO.

- 180 Ringing (Timbrando)

La respuesta 180 Trying es usada para indicar que la INVITE ha sido recibida por el UA y que se esta alertando. Esta respuesta es importante en el trabajo interno con protocolos de telefonía ya que se usa para mensajes como un progreso de ISDN o ISUP. Cuando el agente de usuario UA contesta, inmediatamente, una respuesta 200 OK se envía sin una repuesta 180 RINGING; este escenario es llamado respuesta rápida en telefonía.

Un cuerpo de mensaje en la respuesta podría ser usado para llevar QoS, información de seguridad o para transportar tonos de alerta o animaciones del UAS al UAC. Un UA genera normalmente su propio tono de llamada, a menos de que una cabecera de campo ALERT-INFO (alerta de información) este presente.

- 181 Calls being forwarded (Llamadas siendo remitidas)

La respuesta 181 Calls being forwarded es usada para indicar que la llamada ha sido entregada a otro punto terminal. La respuesta 181 Calls being forwarded se envía cuando esta información puede ser de uso para el visitante o quien realiza la llamada. También, porque una operación de remisión puede tomar mucho para que la llamada sea respondida, esta respuesta da un estado para quien realiza la llamada.

- 182 Call queved (Llamada en cola)

La respuesta 182 Call queved es utilizada para indicar que la INVITE ha sido recibida y será procesada en cola. Puede usarse para indicar el tiempo de espera estimado o

el número de participantes en línea, como se muestra en la Figura 4.9. Un cuerpo de mensaje en la respuesta 182 Call queved puede usarse para llevar música u otro medio de comunicación.

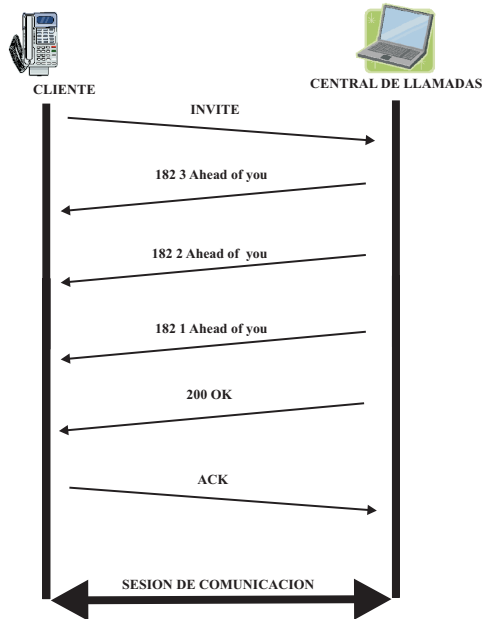


Figura 4.9: Llamada QUEVING

- 183 Session Progress (Progreso de la sesión)

La respuesta 183 SESSION PROGRESS indica que la información acerca del progreso de la sesión CALL STATE (estado de la llamada) se puede presentar en un cuerpo de mensaje ó secuencia multimedia. Sin embargo, a diferencia de una respuesta 180, 181 o 182, esta no transporta ninguna información específica acerca del estado de la INVITE .

La Figura 4.10 muestra el uso de la respuesta 183 SESSION PROGRESS que permite establecer una sesión entre los medios de comunicación antes de que la llamada sea contestada.

Clase 2XX SUCCEs (Exito)

Las respuestas de clase SUCCEs indican que la solicitud es exitosa o ha sido aceptada.

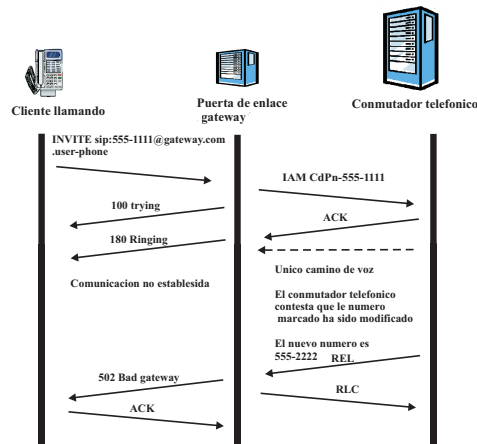


Figura 4.10: Respuesta 183 SESSION PROGRESS

- 200 OK

La respuesta 200 OK tiene dos usos en SIP. Cuando es usado para aceptar una invitación a una sesión, contendrá un cuerpo de mensaje conteniendo las propiedades de medio de comunicación del UAS . Cuando se usa para responder a otra solicitud, indica cumplimiento exitoso o recepción de la solicitud. Además la respuesta detiene las retransmisiones de la solicitud.

En respuesta a una OPTIONS, el cuerpo de mensaje puede contener las capacidades del servidor. Un cuerpo de mensaje también puede presentarse en una respuesta a una solicitud REGISTER . Para una respuesta 200 OK en las solicitudes CANCEL, INFO, MESSAGE, SUBSCRIBE, NOTIFY y PRACK, no es permitido un cuerpo de mensaje.

- 202 Accepted (Aceptado)

La respuesta 202 ACCEPTED [23] indica que el UAS recibió y comprendió la solicitud, pero la misma solicitud no ha sido autorizada o procesada por el servidor. Es comúnmente usada en respuestas para los métodos SUBSCRIBE , REFER y algunos MESSAGE.

Clase 3XX REDIRECTION (Redirección)

Las respuestas de clase REDIRECTION son generalmente enviadas por un servidor SIP actuando como servidor redireccional en respuesta a una INVITE. Sin embargo, un UAS, también puede enviar una respuesta de clase redirección para implementar ciertos tipos de llamadas remitiendo características.

- 300 Multiple Choices (Opciones múltiples)

La respuesta 300 Multiple Choices contiene múltiples cabeceras de campo CONTACT, lo cual indica que el servicio de localización ha retornado múltiples localizaciones posibles para el SIP o SIPS URI en la cabecera REQUEST-URI. Cada una de las opciones deben intentarse en el orden en que fueron escuchadas en la respuesta.

- 301 Moved Permanently (Movido permanentemente)

La respuesta 301 Moved Permanently contiene una cabecera de campo CONTACT con la nueva URI permanente del participante de la llamada. La dirección puede guardarse y usarse en futuras solicitudes de INVITE.

- 302 Moved Temporarily (Movido Temporalmente)

La respuesta 302 Moved Temporarily contiene una URI que es valida actualmente pero no es permanente. Como resultado, la cabecera de campo CONTACT no debe ser almacenada a través de llamadas a menos de que este presente una cabecera de campo EXPIRES, en cuyo caso la localización es valida para la duración del tiempo especificado.

- 305 Use proxy (Utilización de proxy)

La respuesta 305 Use proxy contiene una URI que apunta a un servidor Proxy quien tiene la información de autorización acerca de la llamada. Quien realiza la llamada debe reenviar la solicitud al Proxy para remitirlo. Esta respuesta podría ser enviada por un UAS que esta usando un Proxy al resguardo de llamadas entrantes.

Debido a que el Proxy toma las decisiones por el UAS en la aceptación de la llamada, el UAS solamente responderá a las solicitudes de INVITE que vengan del Proxy de resguardo. Cualquier solicitud de INVITE recibida directamente recibirá automáticamente esta respuesta sin la intervención del usuario.

- 380 Alternative service (Servicio Alternativo)

La respuesta 380 Alternative service retorna una URI que indica el tipo de servicio que la llamada quiere. Un ejemplo sería una redirección a un servidor de correo de voz.

Clase 4XX CLIENT ERROR (Error del cliente)

Las respuestas de clase 4xx se usan por un servidor o UAS para indicar que la solicitud no puede desempeñarse como fue propuesta. La respuesta específica de error del cliente o la presencia de ciertas cabeceras de campo debe indicarle al UAC la naturaleza del error y como puede ser reformulada la solicitud. El UAC no debe volver a proponer la solicitud sin modificar la misma solicitud basándose en la respuesta. La misma respuesta, sin embargo, puede ser probada en otras localizaciones.

Un Proxy bifurcado receptor de una respuesta de clase 4xx no termina la búsqueda. Típicamente, la respuesta de error de cliente requerirá de la intervención del usuario antes de que sea generada una nueva solicitud.

- 400 Bad request (Solicitud errónea)

La respuesta 400 Bad request indica que la solicitud no fue entendida por el servidor. Un ejemplo podría ser una solicitud a la que le faltan las cabeceras de campo requeridas como TO, FROM, CALL-ID o CSEQ. Esta respuesta también es utilizada si un UAS recibe múltiples solicitudes de INVITE, no retransmisiones, del mismo CALL-ID.

- 401 Unauthorized (No autorizado)

La respuesta 401 Unauthorized indica que la solicitud necesita al usuario para realizar la autenticación. Esta respuesta es generalmente enviada por un UA, ya que la respuesta

407 PROXY AUTHENTICATION REQUIRED (autenticación del Proxy requerida) es enviada por un Proxy que necesita autenticación.

La excepción es un registro de servidor, el cual envía una respuesta 401 unauthorized a un mensaje REGISTER que no contiene las credenciales apropiadas. Un ejemplo de la respuesta 401 Unauthorized se muestra en la Tabla 4.25.

```
SIP/2.0 401 Unauthorized
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
```

Tabla 4.25: Ejemplo de la respuesta 401

- 402 Payment required (Pago requerido)

La respuesta 402 Payment required es una función reservada para definiciones futuras en el protocolo SIP. Podría ser usado para negociar cargos de realización de llamada.

- 403 Forbidden (Prohibido)

La respuesta 403 Forbidden se utiliza para negar alguna solicitud sin darle ningún recurso al visitante. Se envía cuando el servidor ha entendido la solicitud, encuentra la solicitud para ser correctamente formulada, pero no la abastece. Esta respuesta se usa cuando es necesaria una autorización.

- 404 Not found (No encontrado)

La respuesta 404 Not found indica que el usuario identificado por la SIP URI o SIPS URI en la cabecera REQUEST-URI no puede ser localizada por los servidores o que el usuario no esta conectado actualmente con el agente de usuario UA.

- 405 Method not allowed (Método no permitido)

La respuesta 405 Method not allowed indica que el agente de usuario UA o el servidor han recibido y entendido una solicitud pero no está dispuesto a cumplir la solicitud.

Un ejemplo sería una solicitud REGISTER enviada a un agente de usuario. Una cabecera de campo ALLOW debe estar presente para informar UAC que métodos son aceptables. Esto es diferente del caso de un método desconocido, en el cual se regresa una respuesta 501 NOT IMPLEMENTED (no implementado).

- 406 Not acceptable (No aceptable)

La respuesta 406 Not acceptable indica que la solicitud no puede proceder debido a un requerimiento en el mensaje de solicitud. La cabecera de campo ACCEPT (aceptar), en la solicitud, no contenía ninguna opción soportada por el UAS.

- 407 Proxy authentication required (Autenticación de proxy requerida)

La solicitud 407 Proxy authentication enviada por un Proxy indica que el agente de usuario cliente UAC primero debe autenticarse por sí mismo con el Proxy antes de que la solicitud pueda ser procesada.

La respuesta contendrá información acerca del tipo de identificaciones requeridas por el Proxy en la cabecera de campo PROXY- AUTHENTICATE (autenticación del Proxy). Al contrario de HTTP, la respuesta 407 Proxy authentication no puede ser usada por un Proxy para autenticar a otro Proxy.

La Tabla 4.26 muestra un ejemplo de este tipo de respuesta.

- 408 Request timeout (Solicitud interrumpida)

La respuesta 408 Request timeout se envía cuando una cabecera de campo EXPIRES está presente en una solicitud INVITE y el periodo de tiempo especificado ha terminado. La solicitud puede reintentarse en cualquier momento por el agente de usuario cliente UAC, tal vez con un periodo de tiempo más grande en la cabecera de campo EXPIRES o sin cabecera de campo EXPIRES en absoluto.

```
SIP/2.0 407 Proxy Authorization Required
via: SIP/2.0/UDP discrete.sampling.org:5060;branch=z9hG4bK6563;
received=65.64.140.198
From: Shannon <sip:shannon@sampling.org>;tag=59204
To: Schockley <sip:shockley@transistor.com>;tag=142334
Call-ID: adf8gasdd7fld@discrete.sampling.org
CSeq: 1 INVITE
Proxy-Authenticate: Digest realm="sampling.org" , qop="auth" ,
  nonce="9c8e88df84dfce4341ae6cbe5a359" ,
  opaque="", stale=FALSE, algorithm=MD5
Content-Length: 0
```

Tabla 4.26: Ejemplo de respuesta 407

- 409 Conflicts (Conflictos)

La respuesta 409 Conflicts ha sido removida de la RFC 2543. Indica que la solicitud no puede ser procesada debido a conflictos en la misma solicitud. Esta respuesta es usada por el registrador para rechazar un registro con conflictos en los parámetros.

- 410 Gone (Acabado)

La respuesta 410 Gone es similar a la respuesta 404 NOT FOUND pero contiene la sugerencia de que el usuario solicitado no estará disponible en esta ubicación en el futuro. Esta respuesta podría ser usada por un proveedor de servicios cuando un usuario cancelara su servicio.

- 411 Length Required (Extensión requerida)

La respuesta 411 Length Required ha sido removida de RFC 3261 pero es definida en RFC 2543. Esta respuesta puede ser utilizada por un Proxy para rechazar una solicitud conteniendo un cuerpo de mensaje pero no la cabecera de campo CONTENT-LENGTH (extensión del contenido).

- 413 Request entity too large (Entidad de solicitud demasiado grande)

La respuesta 413 Request entity too large puede ser utilizada por un Proxy para rechazar una solicitud que tiene un cuerpo de mensaje que es demasiado grande.

- 414 Request-URI too long (Request-URI demasiado grande)

La respuesta 414 Request-URI too large indica que la REQUEST-URI (recurso de identificación uniforme de la solicitud) fue muy grande y no puede ser procesada correctamente. No hay un número máximo de extensión definido para una REQUEST-URI en un documento de estandarización de SIP.

- 415 Unsupported media type (Tipo de medio de comunicación no soportado)

La respuesta 415 Unsupported media type enviada por un UA indica que el tipo de medio de comunicación contenido en la solicitud de INVITE no es soportable. Por ejemplo, una solicitud para una video conferencia de una Gateway en PSTN que solamente maneja llamadas telefónicas resultará en esta respuesta. La respuesta podría contener cabeceras de campo para ayudar al agente de usuario cliente UAC a reformular la solicitud.

- 416 Unsupported URI scheme (Esquema de URI no soportado)

La respuesta 416 Unsupported URI scheme es nueva para la RFC 3261 y es utilizada cuando un UAC usa un esquema de URI en una REQUEST-URI (URI de la solicitud) que el UAS no entiende.

- 420 Bad extension (Extensión errónea)

La respuesta 420 Bad extension indica que la extensión especificada en la cabecera de campo REQUIRE (requerido) no es soportada por el Proxy o el UA.

La respuesta debe contener la cabecera de campo SUPPORTED listando las extensiones que son soportadas. El UAC debe volver a proponer la misma solicitud sin la extensión en la cabecera de campo REQUIRE o proponer la solicitud a otro Proxy o UA.

- 421 Extension required (Extensión requerida)

La respuesta 421 EXTENSION REQUIRED indica que un servidor necesita una extensión para procesar la solicitud que no fue presentada en la cabecera de campo SUPPORTED (soportado) en la solicitud.

La extensión requerida debe estar listada en la cabecera de campo REQUIRED en la respuesta. El cliente debe reintentar la solicitud adicionando la extensión a la cabecera de campo SUPPORTED o intentar la solicitud en un servidor diferente que pueda no necesitar la extensión.

- 422 Session timer interval too small (Intervalo de tiempo de la sesión demasiado pequeño)

La respuesta 422 Session timer interval too small [27] es usado para rechazar una solicitud conteniendo una cabecera de campo SESSION-EXPIRES (vencimiento de la sesión) con un intervalo demasiado pequeño.

La capacidad de rechazar duraciones cortas es importante para prevenir el tráfico excesivo de re-INVITE o de UPDATE. El intervalo mínimo permitido es indicado en la cabecera de campo requerida MIN-SE (intervalo mínimo del vencimiento de la sesión). El solicitante puede reintentar la solicitud sin la cabecera de campo SESSION-EXPIRES o con un valor menor o igual a el especificado.

- 423 Interval too brief (Intervalo demasiado breve)

La repuesta 423 Interval too brief es regresado por un registrante que esta rechazando un registro porque el tiempo de expiración de la solicitud en uno o más CONTACTS es demasiado breve. La respuesta puede contener una cabecera de campo MIN-EXPIRES listando el intervalo mínimo de expiración que el registrante aceptará.

- 428 Use authentication token (Utilizar marca de autenticación)

La respuesta 428 Use authentication token [28] es usada por un UAS que necesita el uso de un AIB [29].

- 480 Temporarily unavailable (Temporalmente no disponible)

La respuesta 480 Temporarily unavailable indica que la solicitud a alcanzado el destino correcto, pero la llamada no esta disponible por alguna razón. La respuesta debe contener una cabecera RETRY-AFTER (reintentar después) indicando cuando la solicitud puede ser capaz de cumplir.

Por ejemplo, la respuesta 480 Temporarily unavailable podría ser enviada cuando un teléfono tiene su tono de timbre apagado o ha sido presionado un botón de "no molestar". Esta respuesta también puede ser enviada por un servidor de redirección.

- 481 Dialog/transaction does not exist (Transacción de diálogo no existe)

La respuesta 481 Dialog/transaction does not exist indica que una respuesta referenciada a una llamada existente, o transacción, ha sido recibida por lo cual el servidor no lo ha registrado o mostrado un estado de la información.

- 482 Loop detected (Ciclo detectado)

La respuesta 482 Loop detected indica que la solicitud ha sido ciclada y ha sido rutada de regreso al Proxy que previamente remitió la solicitud. Cada servidor que remite una solicitud le adiciona una cabecera VIA (vía) con su dirección a la cabeza de la solicitud.

El parámetro BRANCH (sección o sucursal) es también adicionado a la cabecera VIA, la cual es una función mezclada de la REQUEST-URI y de TO/ FROM/CALL-ID y el número CSEQ. La razón para que el parámetro BRANCH (sección o sucursal) deba ser verificado es para permitir que una solicitud sea rutada de regreso a un Proxy.

- 483 Too many hops (Demasiados saltos)

La respuesta 483 Too many hops indica que la solicitud ha sido remitida el número máximo de veces que estaba configurado por la cabecera MAX-FORWARDS (remisiones máximas) en la solicitud.

Esto es indicado por el receptor de la cabecera MAX-FORWARDS=0 en una solicitud. En la Tabla 4.27 se muestra el ejemplo de la respuesta 483, el UAC incluye la cabecera MAX-FORWARDS=4 en la solicitud REGISTER. Un Proxy recibiendo esta solicitud generará cinco intentos y después una respuesta 483.

```
REGISTER sip:registrar.timbuktu.tu SIP/2.0
via: SIP/2.0/UDP 201.202.203.204:5060:branch=z9hG4bK45347.1
via: SIP/2.0/UDP 198.20.2.4:6128:branch=z9hG4bK917a4d4.1
via: SIP/2.0/UDP 18.56.3.1:5060:branch=z9hG4bK7154.1
via: SIP/2.0/TCP 101.102.103.104:5060:branch=z9hG4bKa5ff4d3.1
Via: SIP/2.0/UDP 168.4.3.1:5060:branch=z9hG4bK676746
To: sip:explorer@geographic.org
From: <sip:explorer@geographic.org>;tag=341323
Call-ID: 67483010384@168.4.3.1
CSeq: 1 REGISTER
Max-Forwards: 0
Contacto sip:explorer@national.geographic.org
Content-Length: 0

SIP/2.0 483 Too Many Hops
Via: SIP/2.0/UDP 201.202.203.204:5060:branch=z9hG4bK45347.1
Via: SIP/2.0/UDP 198.20.2.4:6128:branch=z9hG4bK917a4d4.1
Via: SIP/2.0/UDP 18.56.3.1:5060:branch=z9hG4bK7154.1
Via: SIP/2.0/TCP 101.102.103.104:5060:branch=z9hG4bKa5ff4d3.1
Via: SIP/2.0/UDP 168.4.3.1:5060:branch=z9hG4bK676746
To: <sip:explorer@geographic.org>;tag=a5642
From: <sip:explorer@geographic.org>;tag=341323
Call-ID: 67483010384@168.4.3.1
CSeq: 1 REGISTER
Content-Length: 0
```

Tabla 4.27: Ejemplo de la respuesta 483

- 484 Address incomplete (Dirección incompleta)

La respuesta 484 Address incomplete indica que la dirección en la REQUEST-URI no esta completa. La respuesta 484 Address incomplete podría ser usada en un escenario de marcado en el trabajo interno de una PSTN en donde los dígitos son reunidos y enviados hasta que el número de teléfono completo es ensamblado por una Gateway y rutados.

- 485 Ambiguous (Ambiguo)

La solicitud 485 Ambiguous indica que la REQUEST-URI fue ambigua y debe ser aclarada en orden para ser procesada. Esto ocurre si el nombre de usuario coincide con un número de registros. Si son regresadas las opciones posibles de la búsqueda en la

cabecera de campo CONTACT, entonces esta respuesta es similar a la respuesta 300 MULTIPLE CHOICES.

Sin embargo, son ligeramente diferentes, ya que las respuestas de clase 3xx regresan opciones equivalentes para el mismo usuario, pero las respuestas de clase 4xx regresan alternativas que pueden ser para diferentes usuarios.

La respuesta 3xx puede ser procesada sin intervención humana, pero esta respuesta 4xx necesita una opción del visitante a la conversación, lo cual es debido a que esta clasificada como una respuesta de clase de error el cliente.

- 486 Busy here (Ocupado aquí)

La respuesta 486 Busy here es usada para indicar que el UA no puede aceptar la llamada en esta localización. Sin embargo, esto es diferente de la respuesta 600 BUSY EVERYWHERE (ocupado en cualquier lugar), que indica que la solicitud no debería intentarse en ningún sitio.

En general, una respuesta 486 Busy here es enviada por un UAS a menos de que conozca definitivamente que el usuario no puede ser contactado. Esta respuesta es equivalente al tono BUSY en PSTN.

- 487 Request terminated (Solicitud terminada)

La respuesta 487 Request terminated puede ser enviada por un UA que ha recibido una solicitud de CANCEL para una solicitud de INVITE pendiente. Una respuesta 200 OK se envía para confirmara la recepción al CANCEL y una 487 Request terminated se envía en respuesta a una INVITE.

- 488 Not acceptable here (No aceptable aquí)

La respuesta 488 Not acceptable here indica que algunos aspectos de la sesión propuesta no son aceptables y puede contener una cabecera de campo WARNING (advertencia) indicando la razón exacta. Esta respuesta tiene un significado similar a la respuesta 606 NOT ACCEPTABLE, pero solamente aplica a una localización y puede no ser verdad globalmente como la respuesta 606 lo indica.

- 489 Bad event (Evento equivocado)

La respuesta 489 Bad event [23] se utiliza para rechazar una solicitud de suscripción o una notificación conteniendo un paquete de EVENT que es desconocido o no soportado por el UAS. El código de respuesta es utilizado también para rechazar una solicitud de suscripción que no especifica un paquete EVENT, asumiendo que el servidor no soporta el protocolo PINT.

- 491 Request Pending (Solicitud pendiente)

La respuesta 491 Request Pending se usa para resolver re-INVITEs accidentales simultaneas por ambas partes en un diálogo. Puesto que ambas INVITEs buscan cambiar el estado de la sesión, no pueden ser procesadas al mismo tiempo.

Mientras un agente de usuario esta esperando una respuesta final a una re-INVITE, cualquier solicitud de re-INVITE recibida debe ser contestada con este código de respuesta. Esto es análogo a la condición de *çlareo*.^{en} telefonía en la cual ambos extremos toman un troncal al mismo tiempo. Se muestra un ejemplo de la respuesta 491 Request Pending en la Figura 4.11.

- 493 Request undecipherable (Solicitud indescifrable)

La respuesta 493 Request undecipherable se utiliza cuando un cuerpo de mensaje S/MIME no puede ser descifrado porque la llave pública no esta disponible. Si el UAS no soporta S/MIME, no serán presentados cuerpos de mensaje en la respuesta. Si el UAS soporta S/MIME, la respuesta contendrá un cuerpo de mensaje conteniendo una llave pública apropiada para que el UAC use la encriptación S/MIME.

Clase 5XX SERVER ERROR (Errores del servidor)

Las respuestas de clase 5XX son usadas para indicar que la solicitud no puede ser procesada debido a un error con el servidor. La respuesta puede contener una cabecera de campo llamada RETRY-AFTER (reintentar después de) si el servidor anticipa estar disponible con un periodo de tiempo especifico. La solicitud puede intentarse en otras localizaciones porque no hay errores indicados en la solicitud.

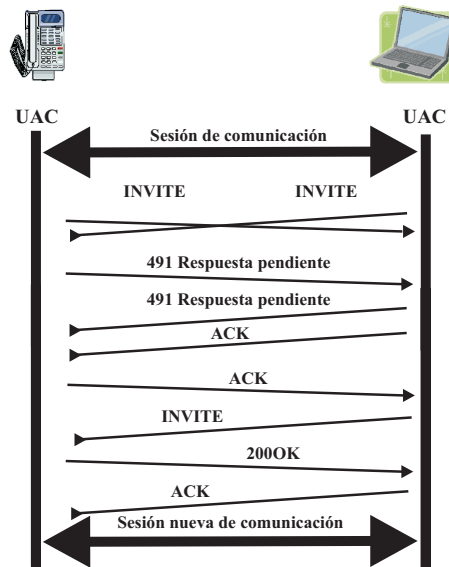


Figura 4.11: Resolución de re-INVITE simultáneas

- 500 Server internal error (Error interno del servidor)

La respuesta 500 Server internal error en el servidor indica que el servidor ha experimentado algún tipo de error que le impide procesar la solicitud. La frase razonable puede usarse para identificar el tipo de falla. El cliente puede reintentar la solicitud de nuevo en el mismo servidor después de algunos segundos.

- 501 Not implemented (No implementado)

La respuesta 501 Not implemented indica que el servidor está imposibilitado para procesar la solicitud porque no es soportada. Esta respuesta puede usarse para rechazar una solicitud conteniendo un método desconocido. Sin embargo, un Proxy remitirá una solicitud conteniendo un método de solicitud desconocido. Así, un Proxy remitirá una solicitud SELF-DESTRUCT (autodestrucción), asumiendo que el UAS generará esta respuesta si el método es desconocido.

- 502 Bad gateway (Puerta de enlace errónea)

La respuesta 502 Bad gateway es enviada por un Proxy que está actuando como una

GATEWAY para otra red de trabajo e indica que algún problema en la otra red de trabajo esta impidiendo que la solicitud sea procesada.

- 503 Service unavailable (Servicio no disponible)

La respuesta 503 Service unavailable indica que el servicio solicitado esta temporalmente no disponible. La solicitud puede reintentarse después de unos segundos o después del vencimiento de la cabecera de campo RETRY-AFTER.

En lugar de generar la respuesta 503 Service unavailable, un servidor cargado solo rechaza la conexión. Este código de respuesta es importante cuando se recibe activado un nuevo DNS mejorando la localización servidor de respaldo para obtener el servicio deseado.

- 504 Gateway timeout (Puerta de enlace interrumpida)

La respuesta 504 Gateway timeout indica que la solicitud fallo debido a una interrupción encontrada en la otra red de trabajo a la cual el GATEWAY conecta. Es una respuesta de clase error en el servidor porque la llamada esta fallando debido a una falla del servidor en acceder a los recursos fuera de la red de trabajo SIP.

- 505 Version not supported (Versión no soportada)

La respuesta 505 Version not supported indica que la solicitud fue rechazada por el servidor debido al número de versión SIP de la solicitud. La semántica detallada de esta respuesta no ha sido aún definida porque hay solamente una versión de SIP (versión 2.0) actualmente implementada. Cuando lo números de las versiones adicionales sean implementadas en el futuro, el mecanismo de distribución con múltiples versiones de protocolos necesitará ser detallada.

- 513 Message too large (Mensaje demasiado grande)

La respuesta 513 Message too large es utilizado por un UAS para indicar que el tamaño de la solicitud fue demasiado grande para ser procesada.

Clase 6XX GLOBAL ERROR (Error global)

Las respuestas de clase 6XX indican que el servidor sabe que la solicitud fallará donde quiera que se pruebe. Como resultado, la solicitud no debe enviarse a otra ubicación. Solamente un servidor que definitivamente tenga conocimiento de la identidad del usuario por la REQUEST-URI en cualquier posible instancia deberá enviar una respuesta de clase error global. De otro modo, deberá enviarse una respuesta de clase error del cliente. La cabecera RETRY-AFTER puede usarse para indicar cuando la solicitud sea exitosa.

- 600 Busy everywhere (Ocupado en cualquier lugar)

La respuesta 600 Busy everywhere es versión definitiva de la respuesta de error del cliente 486 Busy here. Si hay una posibilidad de que la llamada a la REQUEST-URI especificada pueda ser contestada en otras ubicaciones, esta respuesta no será enviada.

- 603 Decline (Declinar)

La respuesta 603 Decline tiene el mismo efecto que la respuesta 600 busy everywhere pero no da afuera ninguna información acerca del estado de la llamada o del servidor. La respuesta 603 Decline podría indicar que los participantes de la llamada están ocupados o simplemente no quieren aceptar la llamada.

- 604 Does not exist anywhere (No existe en ningún lugar)

La respuesta 604 Does not exist anywhere es similar a la respuesta 404 not found pero indica que el usuario en la REQUEST-URI no puede ser hallado en ningún lugar. Esta respuesta debe ser enviada solamente por un servidor que tiene acceso a toda la información acerca del usuario.

- 606 Not acceptable (No aceptable)

La respuesta 606 Not acceptable puede ser usada para implementar algunas capacidades de negociación de sesión en SIP. Indica que algún aspecto de la sesión deseada no es aceptable para el UAS y como resultado no puede ser establecida la sesión.

4.6. Comparación con otras tecnologías

En este punto se compara SIP con otro protocolo de señalización para telefonía IP: la recomendación ITU H.323, identificada como Comunicación Multimedia Basada en Paquetes.

4.6.1. Introducción a H.323

H.323 [30] es una recomendación que abarca todos los aspectos de la comunicación multimedia sobre redes de trabajo de paquetes. Es parte de los protocolos H.32x que describen la comunicación multimedia sobre ISDN, banda ancha ATM, telefonía PSTN y redes IP de paquetes como lo muestra en la Tabla 4.28.

Desarrollado originalmente para video conferencias sobre un solo segmento de LAN, el protocolo ha sido extendido para cubrir problemas generales de telefonía sobre Internet. La primera versión fue aprobada por la ITU en 1996 y fue adoptada pronto por redes de telefonía IP porque no había otros estándares.

La versión 2 fue adoptada en 1998 para reparar algunos de los problemas y limitaciones de la versión 1. La versión 3 fue adoptada en 1999 e incluía modificaciones y extensiones habilitando comunicaciones sobre grandes redes.

La versión 4 fue adoptada en el 2000 con cambios mayores al protocolo. La versión 5 esta actualmente bajo revisión de la ITU. Actualmente, muchos sistemas desplegados usan la versión 2. H.323 ha sido designado para ser compatible hacia atrás, entonces, por ejemplo, una versión 1 conforme a un punto terminal puede comunicarse con un Gatekeeper versión 3 y un punto terminal versión 4.

H.323 hace referencia a un número de otros protocolos de la ITU y de IETF para complementar específicamente el entorno. Cada elemento de la red de trabajo es definido y estandarizado. La Figura 4.12 muestra los elementos principales: terminales, gatekeeper, gateway y MCU. Las terminales, los gateways y los MCUs son equipos dispositivos al final de la red de trabajo, a menudo llamados puntos terminales.

Un punto terminal origina y termina secuencias de medios de comunicación como

Protocolo	Título
H.320	Comunicación sobre redes RDSI
H.321	Comunicación sobre redes de ancho de banda ATM
H.322	Comunicación sobre redes LAN con certificación QoS
H.323	Comunicación sobre redes LAN sin certificación QoS(IP)
H.324	Comunicación sobre redes PSTN (modem)

Tabla 4.28: Familia de estándares ITU 32x

pueden ser audio, video, datos o la combinación de los tres. Como mínimo, todos los puntos terminales de H.323 deben soportar la transmisión de audio básica G.711 PCM. El soporte de video y datos es opcional. Un Gatekeeper H.323 es un servidor que controla una zona, el cual es el dominio administrativo más pequeño en H.323.

Si esta presente un Gatekeeper, todos los puntos terminales que están en la zona deben registrarse y diferir al Gatekeeper en la autorización de decisiones para colocar o aceptar una llamada. También proporciona servicios a terminales en una zona, tales como ubicación de Gateways, interpretación de direcciones, administración del ancho de banda, implementación de características y registro.

Un Gatekeeper no es un elemento requerido en una red de trabajo H.323, pero las capacidades de una terminal sin uno son severamente limitadas. Un Gateway es otro elemento opcional en una red H.323. Une la red H.323 con otra red de otro protocolo como el PSTN. Un MCU proporciona servicios de conferencias para terminales. Algunos de los protocolos referenciados por H.323 se muestran en la Tabla 4.29.

H.225 se utiliza para el registro, admisión y estado RAS, que se utiliza para la comunicación de una terminal con un Gatekeeper. Un subconjunto modificado de Q.931 se usa para configurar señalización de llamadas entre terminales. (El uso H.323 de Q.931 no es compatible con Q.931 como se usa en una red digital de servicios integrados ISDN).

H.245 se usa para la señalización de control o negociación de los medios de comunicación y el intercambio de capacidades entre terminales. TI20 se usa para la comunicación

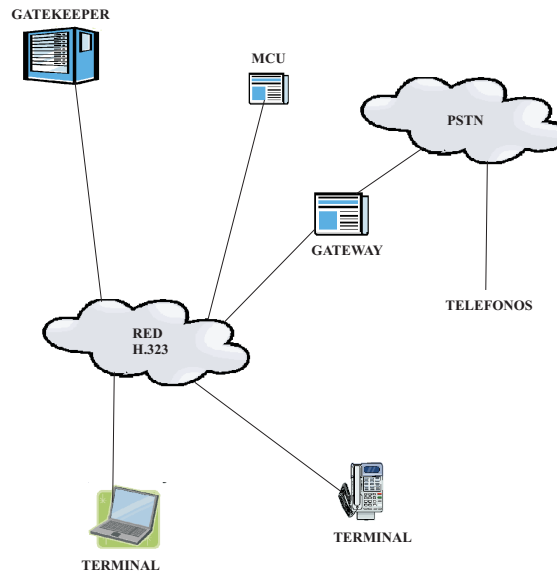


Figura 4.12: Elementos principales de una red H.323

gráfica multipunto. Los códigos de audio de H.323 son especificados en las series G.7xx de la ITU.

Los códigos de video son especificados en las series H.26x. H.323 también hace referencia a dos protocolos de IETF, RTP y RTCP, para el transporte. La recomendación H.235 cubre la privacidad y encriptación, mientras que H.450 cubre los servicios suplementarios como los comúnmente encontrados en PSTN por ejemplo remisión de llamadas, llamada en espera y otros.

La Figura 4.13 muestra el flujo de llamada básico incluyendo a dos terminales y un Gatekeeper. El flujo muestra la interacción entre varios elementos y varios protocolos usados para establecer la sesión. La llamada comienza con un intercambio de mensajes H.225 RAS entre la terminal llamando y el Gatekeeper.

Todos los mensajes RAS son transportados utilizando UDP. Se asume que ambas terminales han sido registradas con el Gatekeeper usando un mensaje RRQ. La terminal llamando envía un mensaje ARQ al Gatekeeper conteniendo la dirección de la terminal que esta llamando y el tipo de sesión deseada.

La dirección podría especificarse como un alias en H.323, el número de teléfono E.164, la dirección e-mail o una URL. El Gatekeeper conoce todas las llamadas en su zona de

Protocolo	Descripción
H.225	Registro, admisión y estado
H.245	Control de señalización
T.120	Comunicación gráfica multipunto
G.7xx	Codecs de audio
H.26x	Codecs de video
RTP	Protocolo de transporte en tiempo real
RTCP	Control de protocolo TCP
H.235	Privacidad y encriptacion
H.450	Servicios suplementarios

Tabla 4.29: Protocolos referidos por H.323

control, decide si el usuario esta autorizado para hacer la llamada y si hay suficiente ancho de banda u otros recursos disponibles.

En el ejemplo planteado, hay bastante ancho de banda, así que el Gatekeeper le permite a la llamada continuar enviando un mensaje de confirmación de admisión (ACF). El mensaje ACF indica a la terminal llamando que rute el mensaje al punto terminal o, que se va a usar, el intercambio directo de mensajes de señalización de llamada H.225 con la terminal llamando.

Alternativamente, el Gatekeeper puede requerir de la señalización rutada, en donde el Gatekeeper actúa como un Proxy y remite todos los mensajes de señalización entre las terminales. También ha interpretado el destino en el mensaje ARQ dentro de una dirección de transporte que fue retornada en la ACF.

La terminal llamando esta ahora habilitada para abrir una conexión TCP a la terminal de llamada utilizando la dirección de transporte retornada en el mensaje ACF y envía un mensaje de estructuración Q.931 a esta misma.

La terminal de llamada contesta con una respuesta CALL PROCEEDING (proceder con la llamada) a la terminal llamando. La terminal de llamada debe también obtener permiso del Gatekeeper antes de aceptar la llamada, así que le envía un ARQ al Gatekeeper.

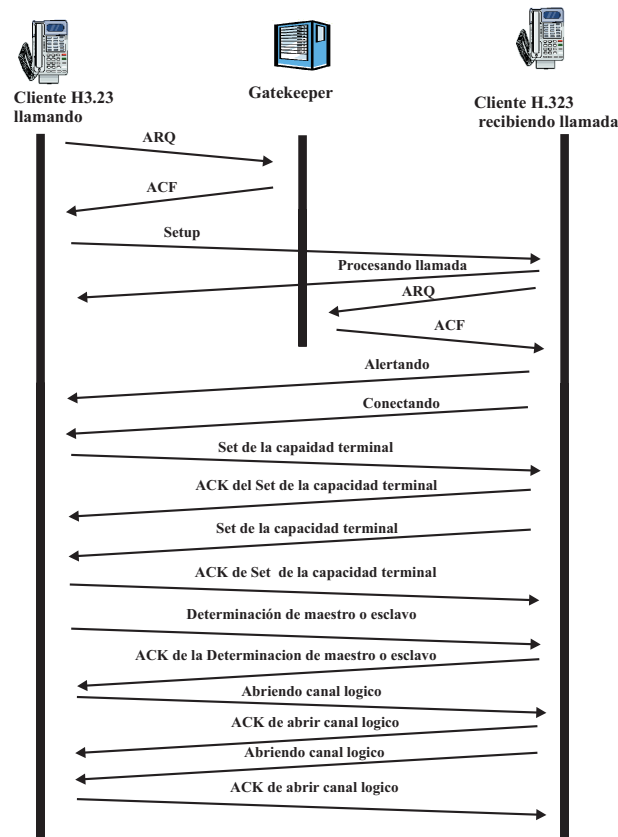


Figura 4.13: Flujo de llamada H.323

Cuando recibe la ACF del Gatekeeper, la terminal de llamada comienza a alertar al usuario y envía un mensaje de alerta a la terminal llamando. Cuando el usuario en la terminal llamando contesta, se envía un mensaje de CONNECT (conectar). No hay reconocimiento de mensajes porque todos estos mensajes son enviados usando TCP, el cual proporciona transporte confiable.

Estos mensajes de señalización de llamada usados en H.323 son un subconjunto de la recomendación Q.931 que cubre los canales de señalización D en ISDN. La siguiente fase es la negociación de conexión, la cual es manejada por los mensajes de control de señalización de H.245. Una segunda conexión de TCP entre las dos terminales se abre por la terminal llamando utilizando el número de puerto seleccionado por la terminal de llamada y regresada en el mensaje de CONNECT.

El mensaje enviado CAPABILITY SET (establecimiento de las capacidades) de la terminal contiene las capacidades de medios de comunicación de la terminal llamando,

listando los códigos soportados. Esto es conocido con la respuesta `TERMINAL CAPABILITY SET ACK` (confirmación de recepción de las capacidades de terminal establecidas) de la terminal de llamada.

La terminal de llamada también envía un mensaje de `TERMINAL CAPABILITY SET` (establecimiento de capacidades de la terminal) conteniendo sus capacidades de los medios de comunicación, las cuales se reciben en una respuesta `TERMINAL CAPABILITY SET ACK`.

El protocolo H.323 requiere que una terminal sea seleccionada como el master (maestro) y la otra como slave (esclavo). Esto se logra usando el intercambio de mensajes `MASTER/SLAVE DETERMINATION` (determinación de maestro y esclavo) entre las terminales. El mensaje contiene el tipo de terminal de la terminal (es decir maestro o esclavo) y un número aleatorio.

Los tipos de terminales son jerárquicos, lo cual determina primero al master (maestro). Si el tipo de terminal es el mismo, el número aleatorio determina al slave (esclavo). El mensaje es conocido como `MASTER/SLAVE DETERMINATION ACK` (confirmación de recepción de la determinación de maestro o esclavo).

La fase final del establecimiento de la llamada es la apertura de dos canales lógicos entre las terminales. Estos canales son usados para la estructuración y control de los canales de los medios de comunicación.

El mensaje H.245 `OPEN LOGICAL CHANNEL` (apertura de canal lógico) enviado en la conexión de control de señalización H.245 contiene el tipo de medio de comunicación deseado, incluyendo el código que ha sido determinado del intercambio de capacidades.

También contiene un par de direcciones para las secuencias de RTP y RTCP. El mensaje es conocido como `OPEN LOGICAL CHANNEL ACK` (confirmación de recepción de apertura de canal lógico).

Ahora, las terminales inician el envío de paquetes de medios de comunicación RTP y también de paquetes de control de RTCP usando la dirección IP y el número de puerto intercambiado en los mensajes `OPEN LOGICAL CHANNEL`.

La Figura 4.14 muestra una secuencia de terminación de llamada, en la cual cualquier

terminal puede iniciar. La terminal de llamada envía un mensaje END SESSION COMMAND (comando de fin de sesión) en el canal de control de señalización H.245.

La otra terminal responde con un mensaje END SESSION COMMAND en el canal de control de señalización H.245, con lo cual ahora puede cerrarse. La terminal de llamada entonces envía un mensaje (DRQ) y recibe un mensaje (DCF) del Gatekeeper. De esta forma, el Gatekeeper sabe que el recurso usado en la llamada ahora se ha liberado.

Después, un mensaje de liberación completa Q.931 se envía en la señalización de conexión de llamada, que puede entonces cerrarse. Finalmente, la otra terminal envía una DRQ al Gatekeeper sobre UDP y recibe una respuesta DCF.

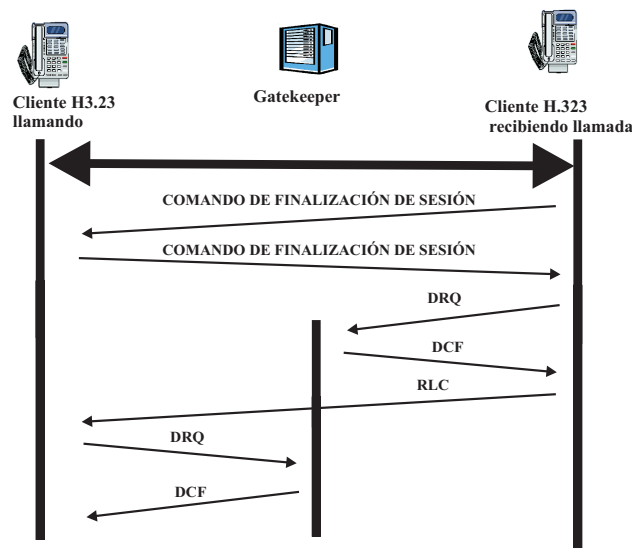


Figura 4.14: Secuencia de terminación de llamada H.323

Los flujos de llamada en las Figuras 3.31 y 3.32 muestran señalización directa al punto terminal, en donde la terminal llamando abre una conexión TCP a la terminal de llamada e intercambia mensajes de H.225.0 y H.245.

En la respuesta ACF a la terminal llamando, el Gatekeeper puede necesitar de señalización de ruteo de Gatekeeper, en donde los canales de señalización de llamada y de señalización de control son abiertos con el Gatekeeper, quien entonces abre los canales con la terminal de llamada.

4.6.2. Comparación de H.323 con SIP

SIP y H.323 fueron desarrollados para diferentes propósitos por cuerpos de estándares con diferentes requerimientos. H.323 fue desarrollado por la ITU. Su diseño e implementación reflejan su antecedente de PSTN, utilizando codificación binaria y reutilizando partes de la señalización ISDN.

SIP, del otro lado, fue desarrollado por la IETF con una perspectiva de Internet, diseñado para ser escalable sobre el Internet y trabajar en una vía de dominios internos utilizando el establecimiento completo de utilidades y funciones de Internet.

Mientras H.323 fue desplegado pronto en VoIP y aplicaciones de video conferencias IP, SIP con su arquitectura de Internet esta ganando impulso y esta emergiendo como el futuro en estándares de señalización para comunicaciones IP , como lo es telefonía IP.

- Diferencias fundamentales

La primera diferencia es en el esquema de codificación usado por el protocolo. SIP es un protocolo basado en texto como lo es HTTP y SMTP, mientras que H.323 utiliza codificación binaria de mensajes ASN.1. La codificación binaria de H.323 puede resultar en tamaño de mensajes más pequeños pero adiciona complejidad para implementaciones.

Un protocolo basado en texto como lo es SIP puede ser fácilmente escrito y no requiere de herramientas para supervisar e interpretar mensajes, un simple paquete descargado de una LAN proporciona la codificación ASCII del mensaje SIP, los cuales pueden ser fácilmente registrados, examinados e inspeccionados.

Pueden escribirse aplicaciones simples para probar y simular el tráfico en SIP. La codificación basada en texto de SIP ha hecho que parezca más amigable a desarrollos de Internet y Web para quienes lo han adoptado en el desarrollo de varias aplicaciones.

Otra diferencia importante es que mientras H.323 es exclusivamente un protocolo de señalización, SIP tiene capacidades tanto de presencia como de mensajería instantánea. La combinación de presencia y señalización en un protocolo usando un esquema de direccionamiento común universal, URI, sería un poderoso controlador en aplicaciones nuevas en el futuro.

También, la combinación de presencia y de mensajería instantánea, hace a SIP un extremadamente poderoso concentrado de protocolos que permiten a un usuario con múltiples puntos terminales móviles localizar y comunicar con otros usuarios con múltiples capacidades. Es por esta razón que los más increíble e innovadores servicios están siendo desarrollados con SIP en lugar de H.323.

Otra importante diferencia está en el soporte de fabricantes y el impulso. SIP ha sido adoptado por algunos participantes importantes en la industria de las telecomunicaciones y PCs. Es ampliamente tomado en cuenta en la industria como en los protocolos de señalización para establecimiento de sesiones sobre IP.

Mientras es inmediatamente posible reemplazar varias redes establecidas con H.323 usadas para servicios básicos como simples llamadas telefónicas, SIP esta siendo adoptado o investigado por todos los integrantes más grandes de la industria, incluso aquellos quienes actualmente tienen intereses establecidos legalmente en sistemas de H.323.

SIP también ha sido adoptado por operadores móviles como la señalización de llamada y protocolo de mensajes instantáneos para la tercera generación de redes de trabajo que están actualmente bajo desarrollo. De igual forma ofrece la promesa de millones de dispositivos inalámbricos habilitados con SIP en los siguientes cinco años. SIP esta también siendo acoplado a redes inalámbricas 802.11 de otros conjuntos de servicios móviles.

Otra diferencia importante es el nivel de seguridad en el protocolo. SIP tal como se define en RFC 3261 tiene un mecanismo de seguridad bastante robusto para proporcionar encriptación, autenticación usando certificados e integración de mensajes end-to-end incluso en la presencia de servidores intermediarios poco confiables.

SIP no necesita desarrollar estas características de seguridad: en su lugar, ya que es un protocolo de Internet, fue capaz de heredar un conjunto de protocolos de seguridad de Internet tales como TLS y S/MIME de manera muy directa. Por ejemplo, el mismo mecanismo de seguridad que hace segura la entrada a información de tarjetas de crédito en una forma de página Web de seguridad es el que permite a SIP proporcionar señalización de seguridad entre servidores.

Aún cuando no es directamente relacionado con SIP, las capacidades de negociación de los medios de comunicación de SDP son bastante diferentes a las que proporciona

H.245 para H.323. H.245 siempre es criticado por su complejidad, pero SDP siempre es criticado adecuadamente por su falta de expresión.

El legado multicast de SDP no proporciona a SIP buenas bases para negociación de medios de comunicación. Como resultado, algunas capacidades de los códigos de medios de comunicación de negociación son muy pobres en redes internas entre implementaciones debido a estos problemas. Mucho del trabajo de las industrias ha sido para arreglar este problema, pero aun no hay a la vista un reemplazo para SDP.

En los años que han coexistido, SIP y H.323 tienen actualmente más similitud en algunas funcionalidades. Esto es natural puesto que cada protocolo ha adoptado algunas características de utilidad para cada uno. Por ejemplo, mientras SIP tiene la forma correcta de iniciar el soporte de DNS y URL, H.323 no inicia con ninguno de ellos pero ha agregado algún soporte para ellos. Otra similitud relacionada a la conferencia es que mientras H.323 tiene el concepto correcto de inicio de una MCU, SIP inicialmente no tiene definidos mecanismos de conferencias.

Sin embargo, el trabajo para definir la señalización equivalente en SIP esta actualmente en marcha. También están en marcha otros trabajos en la IETF para estandarizar las funciones de combinación de los medios de comunicación de una MCU.

De otra manera, los protocolos iniciaron en puntos finales opuestos del espectro y se han movido hacia otros. Por ejemplo, SIP fue inicialmente desplegado exclusivamente con UDP, pero el soporte TCP ha sido más importante en los últimos años. H.323, del otro lado, no podría inicialmente ser usado exclusivamente sobre UDP, pero ahora ha sido extendido para poderse usar sobre este protocolo.

Finalmente, H.323 inició con más especificaciones que SIP, pero como SIP ha crecido y adicionado funcionalidad, ahora ha ganado la guerra de pisapapeles en términos de más páginas de especificaciones de texto.

Capítulo 5

Implementación del servidor SIP

5.1. Introducción

SIP es un protocolo que trabaja directamente en la capa de sesión del modelo de referencia OSI, de tal manera que es posible iniciar, mantener y finalizar sesiones que permiten, entre otras cosas, la comunicación telefónica a través del Internet, mensajería instantánea y comunicación multimedia.

Al ser un protocolo que permite comunicación telefónica significa que trabaja en tiempo real, así que entre otras características debe trabajar en una plataforma confiable. En el presente capítulo se lleva a cabo una comunicación telefónica usando el SIP sobre la plataforma de código abierto Linux y utilizando un servidor de SIP de carácter PBX denominado Asterisk.

5.2. Linux como plataforma para SIP

Linux es la denominación de un sistema operativo y el nombre de un núcleo [31]. Linux es una de las opciones del desarrollo de software libre (y de código abierto), en donde el código fuente está disponible públicamente y cualquier persona, con los conocimientos informáticos adecuados, puede libremente estudiarlo, usarlo, modificarlo y redistribuirlo.

El término Linux es comúnmente utilizado para describir al sistema operativo tipo Unix que utiliza primordialmente filosofía y metodologías libres, también conocido como GNU/Linux, que está estructurado mediante la combinación del núcleo Linux con las bibliotecas y herramientas del proyecto GNU y de muchos otros proyectos y grupos de software (libre o no libre)[32].

Actualmente, una parte considerable de las aplicaciones que conforman un sistema Linux son de origen distinto a GNU (KDE, OpenOffice.org, Firefox, Apache, MySQL, PostgreSQL, Ruby, Perl, Python, entre otros). El proyecto GNU posee desde hace muchos años su propio núcleo en desarrollo, llamado Hurd. Linux brinda una alternativa frente a sistemas operativos no libres como Unix y Windows.

Este núcleo (en inglés kernel), escrito completamente en C con algunas extensiones GNU C, fue desarrollado por el hacker finlandés Linus Torvalds en 1991 en un intento por obtener un sistema operativo libre similar a Unix que funcionara con microprocesadores Intel 80386.

Desde su lanzamiento, Linux ha incrementado su popularidad en el mercado de servidores. Su gran flexibilidad ha permitido que sea utilizado en un rango muy amplio de sistemas de cómputo y arquitecturas, computadoras personales, supercomputadoras, dispositivos portátiles, etc.

Los sistemas Linux funcionan sobre más de 20 plataformas diferentes de hardware; entre ellas las más comunes son las de los sistemas compatibles con PCs x86 y x86-64, computadoras Macintosh, PowerPC, Sparc y MIPS.

5.2.1. Distribuciones de Linux

Las distribuciones Linux, también llamadas distros [33], son colecciones de software que suelen contener grandes cantidades de paquetes además del núcleo Linux. El software que suelen incluir consta de una enorme variedad de aplicaciones, como: entornos gráficos, herramientas ofimáticas, servidores web, servidores de correo, servidores FTP, etc.

Algunas de las distribuciones mas comerciales son:

- ASLinux Desktop (distribución para escritorios de descarga gratuita basada en Debian y KDE).
- Caldera Linux
- Corel Linux (basada en Debian)
- Linspire (basada en Debian, antes Lindows)
- Lycoris Desktop/LX
- Mandriva (fusión de Conectiva y Mandrake)
- Tumix GNU/linux
- Red Hat Linux
- RXART(distribución de linux basada en Debian y orientada al mercado latino comercializada en Perú, Uruguay, Chile y Argentina).
- Red Hat Enterprise Linux
- SUSE Linux (x86/PPC)
- Turbolinux
- Xandros (basada en Corel Linux e inspirada en Debian)
- Yellow Dog Linux (para PPC, basada en Fedora Core PPC)

5.2.2. Distribución SUSE

Linux SUSE es una de las distribuciones más conocidas de Linux a nivel mundial. Entre las principales virtudes de la distribución SUSE se encuentra el que sea fácil de instalar y administrar, ya que cuenta con varios asistentes gráficos para completar diversas tareas, en especial por su herramienta de instalación y configuración YaST.

El nombre SUSE es el acrónimo del alemán "Software- und Systementwicklung", el cual formaba parte del nombre original de la compañía y que se podría traducir como "desarrollo de software y sistemas".

5.3. PBX Asterisk

Asterisk es un completo PBX IP en software [34]. Corre en una amplia variedad de sistemas operativos incluyendo Linux, Mac OS X, OpenBSD, FreeBSD y Sun Solaris, proporciona, además, todas las características que podrían esperarse de un PBX, incluyendo varias características avanzadas que están frecuentemente asociadas a grandes terminales.

La arquitectura de Asterisk esta diseñada para tener máxima flexibilidad y soporte de VoIP en varios protocolos y puede interoperar con la mayoría de los estándares y equipos de telefonía utilizando hardware relativamente no muy caro.

5.3.1. Hardware Soportado

Asterisk no necesita hardware adicional para VoIP. Para interconectarse con equipos de telefonía análogos y digitales, Asterisk soporta un gran número de dispositivos diferentes tales como los teléfonos de Cisco y Avaya, y de manera particular los dispositivos manufacturados por Digium, la empresa creadora de Asterisk.

5.3.2. Características

Las soluciones de telefonía basadas en Asterisk ofrecen un surtido y flexible conjunto de características. Asterisk proporciona tanto las funcionalidades clásicas de un PBX como características avanzadas, las cuales interoperan con los estándares de sistemas de telefonía tradicionales y sistemas de Voz sobre IP (VoIP).

5.3.3. Protocolos soportados

Asterisk soporta un amplio rango de protocolos que usan y transmiten voz sobre interfaces de telefonía tradicionales incluyendo H.323, Protocolo de inicio de sesión (SIP), Protocolo de control de enlace de medios de comunicación (MGCP), Protocolo de control de cliente Skinny (SCCP).

El uso de paquetes de voz le permite a Asterisk enviar datos tales como información URL e imágenes junto con el tráfico de voz, lo que permite una integración avanzada de información.

Asterisk proporciona un núcleo de conmutación central, con cuatro APIs para modular la descarga de aplicaciones de telefonía, dispositivos de interfaz, formato de archivos y codecs. Esto permite una conmutación transparente entre todas las interfaces soportadas y juntar diversas combinaciones de sistemas de telefonía dentro de una sola red de conmutación.

5.3.4. Arquitectura de Asterisk

Asterisk está cuidadosamente diseñado para tener una máxima flexibilidad e interoperabilidad. Así que son definidos en su estructura APIs [35] específicos alrededor de un núcleo avanzado de sistema central PBX.

El núcleo avanzado establece la interconexión interna del PBX, de manera independiente y abstracta de los protocolos específicos, codecs e interfaces de las aplicaciones de telefonía que permiten a Asterisk utilizar cualquier dispositivo y tecnología disponible ahora o en el futuro para mejorar su función esencial que es conectar dispositivos y aplicaciones.

La definición de los APIs facilitan la abstracción de dispositivos y protocolos. Utilizando este sistema de descarga de módulos, el núcleo de Asterisk no tiene que preocuparse acerca de los detalles de como se esta conectando un interlocutor, que codecs están en uso, etc. La función de cada uno de los APIs se muestra a continuación.

API de Canal

El API de canal establece el tipo de conexión con la cual un interlocutor se esta presentando, pudiendo ser una conexión VoIP, ISDN, PRI, señalización de Bits o alguna otra tecnología. Los módulos dinámicos son descargados para asegurar los detalles de las capas mas bajas de estas conexiones.

API de Aplicación

El API de aplicación permite a varios módulos de tareas ejecutarse para realizar varias funciones como conferencias, búsqueda, listado de directorios, correo de voz, transmisión en línea de datos y cualquier otra tarea que el sistema PBX pueda realizar.

API de Traducción de códigos

El traductor de códigos API descarga módulos de codificación para soportar varios formatos de codificación y decodificación de audio tales como GSM, Mu-Law, A-Law e incluso MP3.

API de Formato de archivos

El API de formato de archivo, establece la escritura y lectura de varios formatos de archivos para el almacenamiento de datos en el sistema de archivos. Utilizando estos APIs, Asterisk adquiere una completa abstracción entre sus funciones como un sistema servidor PBX y las variadas tecnologías existentes (o en desarrollo) en el área de la telefonía.

La forma modular (por su composición con módulos) es lo que permite a Asterisk integrarse de la misma forma ya sea con dispositivos de conmutación telefónica implementados actualmente como con las crecientes tecnologías de paquetes de voz.

La habilidad de cargar módulos de codificación también le permite a Asterisk soportar los extremadamente compactos códigos necesarios para paquetes de voz sobre conexiones lentas como un módem telefónico, proporcionando aun gran calidad de audio sobre conexiones menos veloces.

5.4. Instalación de Linux

Es necesario obtener la distribución de Linux deseada, teniendo en cuenta que cualquier distro es un candidato confiable para trabajar con el PBX de Asterisk. Para el servidor

SIP del presente trabajo se utilizó el distro SUSE.

Para comenzar se inserta el CD en la unidad lectora y se reinicia la computadora para comenzar desde el instalador de SUSE. Si esto no funciona y carga Windows, seguramente es porque la Bios de la PC esta configurada para cargar el disco duro en primera instancia; se debe cambiar esta situación desde el SETUP de la computadora y poner como primera opción al CDRom, en segunda al disco duro y en tercera a la disquetera.

Después de iniciar el sistema operativo, dependiendo de la versión del distro, el instalador pide que se inserten algunos datos, como la ubicación geográfica, idioma, zona horaria, tipo de gestor de archivos (KDE ó GNOME) y password de ROOT.

Al finalizar el procedimiento de instalación la PC se reinicia y solicita un usuario y una contraseña. El usuario es root y la contraseña es la misma que estableció durante la instalación. La contraseña insertada debe ser fácil de memorizar.

5.5. Instalación de Asterisk

Asterisk utiliza tres paquetes principales [35]: el programa principal de Asterisk (asterisk), los controladores de telefonía de Zapata (zaptel) y las librerías de PRI (libpri). Para una red totalmente sobre la tecnología VoIP el único requerimiento es el paquete de asterisk.

Los controladores de zaptel son necesarios si se están usando dispositivos físicos análogos o digitales. La librería libpri es técnicamente opcional a menos de que se estén usando interfaces RDSI PRI y se quiera ahorrar un poco de RAM, pero es recomendable que se instale en conjunto con el paquete de zaptel para complementarse.

Otro paquete que se puede instalar es el asterisk-sounds. Mientras Asterisk viene con algunos sonidos predefinidos en la distribución principal de origen, el paquete asterisk-sounds proporcionará algunos más.

Para compilar Asterisk, es necesario instalar el compilador GCC (versión 3.x o posterior) y sus dependencias, esto debido a que la versión 2.96 de GCC puede trabajar solo

por un tiempo pero las versiones futuras no lo soportarán. Asterisk también necesita de bison, un programa generador de análisis que reemplaza a yacc y ncurses para la funcionalidad CLI. La criptografía de librerías en Asterisk necesita de OpenSSL y sus paquetes desarrollados.

Si se quiere utilizar cualquier controlador de dispositivos físicos proporcionados por Zaptel, se necesita instalar el paquete zaptel como tal. Si se esta usando libpri, recomendable, se debe asegurar su instalación antes de instalar asterisk.

5.5.1. Código fuente de Asterisk

El código fuente de Asterisk puede obtenerse del servidor FTP de Digium (empresa diseñadora de Asterisk), localizado en *ftp://ftp.digium.com*. La forma de obtener la versión estable es a través del uso de una conexión a Internet y el comando wget.

Cabe notar que se hará uso del directorio */usr/src/* de SUSE para extraer y compilar el código de Asterisk, por lo que se debe estar seguro de contar con el acceso de root para guardar los archivos en el directorio */usr/src/* y para instalar Asterisk junto con sus paquetes enlazados.

Para obtener el último código fuente estable a través de wget, se introducen los siguientes comandos en la línea de comando de la consola de linux:

```
cd /usr/src/  
  
wget -passive-ftp ftp.digium.com/pub/asterisk/asterisk-1.2.14.tar.gz  
  
wget -passive-ftp ftp.digium.com/pub/asterisk/asterisk-sounds-1.2.1.tar.gz  
  
wget -passive-ftp ftp.digium.com/pub/zaptel/zaptel-1.2.12.tar.gz  
  
wget -passive-ftp ftp.digium.com/pub/libpri/libpri-1.2-4.tar.gz
```

Al obtener el código fuente necesario se tiene al sistema listo para proceder a la extracción del código. Es importante escribir la versión de los archivos que se quieran utilizar, para el servidor del presente trabajo las versiones en los comandos anteriores fué la utilizada.

5.5.2. Extracción del código fuente

Una vez que se ha utilizado el comando `wget` para obtener el código del servidor FTP, se procede a la descompresión del código fuente antes de compilarlo. Si no se descargaron al inicio los paquetes al directorio `/usr/src/` se pueden mover antes de compilarlos, ó, se especifica la ruta completa para su localización.

El siguiente paso es utilizar la aplicación GNU *tar* para extraer el código del archivo comprimido. Utilizando *tar* se extraerán los paquetes y su respectivo código a los directorios correspondientes. La extracción es un proceso que puede ejecutarse a través del uso de los siguientes comandos:

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

```
tar xvf zaptel-1.2.12.tar.gz
```

```
tar xvf libpri-1.2.4.tar.gz
```

```
tar xvf asterisk-1.2.14.tar.gz
```

```
tar xvf asterisk-sounds-1.2.1.tar.gz
```

5.5.3. Compilación de Zaptel

La interfaz Zaptel [35] es un módulo que presenta una capa de abstracción entre los controladores de dispositivos físicos y el módulo Zapata en Asterisk, esta interfaz le permite a los controladores de dispositivos ser modificados sin realizar ningún cambio a la propia fuente de Asterisk.

Los controladores de los dispositivos son usados para comunicarse directamente con lo físico y pasar la información entre Zaptel y el dispositivo telefónico. Para compilar los controladores de telefonía de Zapata se utilizan los siguientes comandos:

```
cd /usr/src/zaptel-1.2.12
```

```
make clean
```

make

make install

5.5.4. Compilación de Libpri

La compilación e instalación de libpri [35] sigue la misma forma que la descrita para zap-
tel. Libpri es utilizada por varios marcadores de dispositivos físicos con Multiplexación
por División de Tiempo (TDM), pero aunque no se tenga instalado el hardware es
recomendable compilar e instalar esta librería.

Se debe compilar e instalar libpri antes de Asterisk, ya que la librería libri será detectada
y utilizada cuando sea compilado Asterisk. Las siguientes líneas muestran los comandos
para el proceso de instalación de esta librería:

cd /usr/src/libpri-1.2.4

make clean

make

make install

5.5.5. Compilación de Asterisk

La librería de Asterisk [35] es compilada con *gcc* a través del uso de la aplicación GNU
make. A comparación de otros programas, no es necesario correr una configuración
definida para Asterisk. Para comenzar a compilar Asterisk, se corren los siguientes
comandos:

cd /usr/src/asterisk-1.2.14

make clean

make

make install

make samples

Al ejecutar el comando *make samples* se podrán instalar los archivos de configuración predefinida. Todos los valores predefinidos son correctos para Asterisk.

5.5.6. Compilación de Asterisk-sounds

Aunque al descargar Asterisk ya viene con ciertos sonidos predefinidos que se pueden utilizar al realizar la configuración del plan de marcado, es importante descargar el paquete de Asterisk-Sounds [35] para hacer de mayor funcionalidad al servidor SIP. Para compilar Asterisk-sounds se ejecutan los siguientes comandos:

cd /usr/src/asterisk-sounds-1.2.1

make clean

make

make install

5.6. Comandos CLI

Asterisk tiene dos componentes internos: un servidor que funciona en segundo plano y un cliente CLI (Interfaz de Línea de Comandos) que supervisa el servidor. Tanto el servidor como el CLI se utilizan usando la orden Asterisk pero empleando distintos argumentos. El CLI binario de Asterisk se encuentra localizado de manera predefinida en */usr/sbin/asterisk* [35].

De tal manera que para explorar todo el rango de argumentos disponibles antes de inicializar el servidor Asterisk de manera completa, se debe ejecutar Asterisk en la línea de comandos de linux con el argumento *-h*:

/usr/sbin/asterisk -h

Mostrando una lista de los argumentos más comunes:

- c** Consola. Permite conectarse a la CLI de Asterisk.
- v** Depuración. Se utiliza para establecer el tipo y nivel de depuración de la salida del CLI.
- g** Respaldo de Nucleo. Si Asterisk fuera dañado inesperadamente, este argumento permitiría crear un archivo de respaldo del nucleo con el último punto estable con gdb.
- r** Remoto. Se utiliza para reconectarse de manera remota con un proceso de Asterisk que aún se esta corriendo.

Para arrancar el servidor Asterisk en modo de depuración (v) y abriendo un cliente en modo de consola (CLI) con el cual se puede monitorear lo que está pasando en el servidor, se ejecuta el siguiente comando:

```
usr/sbin/asterisk -vvvv
```

Es importante, ya que se tiene iniciado el servidor Asterisk, conocer los comandos básicos que se utilizarán en la CLI a lo largo de toda la configuración del PBX SIP.

Para cargar, ó, volver a cargar, cualquier configuración realizada o modificada se utiliza:

```
CLI> reload
```

Para activar el modo de depuración y poder observar y evaluar el proceso que se está ejecutando en SIP se emplea el siguiente comando:

```
CLI> sip debug
```

Si se quiere desactivar el modo de depuración para SIP se ejecuta:

```
CLI> sip no debug
```

Para mostrar el estado de usuarios, peers y canales para SIP se ejecuta cualquiera de los siguientes comando de acuerdo a la solicitud:

```
CLI> sip show users
```

```
CLI> sip show peers
```

```
CLI> sip show channels
```

Es importante mencionar que para los comandos *sip show* el término "peers" define al término compañero en español. En el caso concreto de Asterisk los términos *users* y *peers* como tal se utilizan para clasificar los tipos de conexiones IP al sistema.

5.7. Configuración de SIP a nivel local

El número de ficheros de configuración que se debe modificar para hacer funcionar a un servidor SIP depende del tipo de tecnologías VoIP que se requiera usar en la instalación de manera simultánea. Una de las ventajas de los servidores SIP es que se puede hacer uso tanto de hardware telefónico (análogo o digital) como de software telefónico (softphones).

Para demostrar las ventajas de esta tecnología en la implementación actual se utilizan Softphones por lo que se habrán de configurar dos ficheros principales:

- *etc/asterisk/extensions.conf*

En donde se configuran los planes de llamada (dialplan) que interconectan a los canales y que harán funcional al servidor SIP.

- *etc/asterisk/sip.conf*

Para realizar la configuración de SIP y de los canales de comunicación de tipo SIP (teléfonos y proveedores SIP). De tal manera que para configurar al servidor de manera local se deben tomar en cuenta dos aspectos generales:

A) Configuración de Canales

- En primera instancia se debe configurar el tipo de canal de comunicación que se requiere usar y que para este servidor son canales de tipo SIP. Para entender el concepto de canal de comunicación se debe imaginar un cable. Los canales en telefonía IP no son los cables físicos, sino cables lógicos.
-

- Como Internet permite tener muchas sesiones concurrentes en el mismo cable físico, se pueden definir múltiples canales lógicos que operan simultáneamente en el mismo medio.

B) Desarrollo de un Plan de Marcado

- Aquí se puede definir de que manera van a interactuar cada uno de los canales, asignados a extensiones, entre si.
- Una vez que se ha definido un canal, se garantiza que las conversaciones puedan entrar y salir del PBX; pero además se debe de definir cómo se encaminan cada una de esas conversaciones.
- Se puede estructurar que una llamada entrante desde el servidor se envíe automáticamente a un teléfono IP , ó, definir una conexión entre dos teléfonos IP separados a gran distancia a través de una red inalámbrica.
- Para poder entender un poco más estos conceptos podemos remontarnos a los sistemas de telefonía más antiguos. En esos sistemas, existía una persona (el operador) que era responsable de conectar físicamente los cables telefónicos entre dos terminales. Para que una llamada fluyera entre dos líneas de comunicación (canales) se necesitaba contactar primero con el operador (PBX) e informarle cual era la intención. El fichero de extensiones que se debe configurar en el servidor en la PBX suplanta al operador tradicional.

5.7.1. Determinación de características para conexión a nivel local

Para la implementación del servidor de comunicación utilizando el protocolo SIP se realiza la instalación de un servidor en una red local. En esta red local se tiene un ordenador en el cual se ha instalado el PBX Asterisk que ofrece el servicio de Telefonía IP a dos clientes, también en la misma red, que tienen instalados un teléfono Softphone respectivamente.

Después de completar la instalación y configuración, cada teléfono debe ser capaz de realizar llamadas hacia el servidor y de un cliente a otro. Es importante mencionar que el número de teléfonos Softphones participantes en la red será de acuerdo a las necesidades del lugar implementado.

En la implementación de un PBX de Asterisk se debe tomar en cuenta el tipo de tecnologías que van converger e interoperar a fin de poder facilitar el soporte y mantenimiento. La Tabla 5.1 muestra la información general de cada uno de los clientes de la red local.

CLIENTE	TECNOLOGÍA	EXTENSIÓN
Maquina 1	Teléfono “softphone” usando SIP	110
Maquina 2	Teléfono “softphone” usando SIP	115

Tabla 5.1: Información general de los clientes locales

La más versátil de las características de la tecnología SIP es que permite la interoperabilidad y la convergencia de tecnologías, por lo que para los clientes se están utilizando dos sistemas operativos diferentes.

El cliente MAQUINA 1 tiene la dirección IP 192.168.10.10 y esta trabajando bajo la plataforma WINDOWS XP, utiliza además el softphone X-LITE y el cliente MAQUINA 2 con dirección IP 192.168.10.15 trabaja bajo la plataforma SUSE LINUX y también usa el softphone X-LITE. La Figura 5.1 muestra las especificaciones técnicas de ambos clientes.

Ambos clientes se conectan a través de un switch, utilizando cable de red Ethernet, directamente hacia el PBX Asterisk que tiene la dirección IP 192.168.10.5 por lo que pertenecen a la misma red local. La Figura 5.2 muestra como queda la interconexión de ambos clientes y el PBX Asterisk.

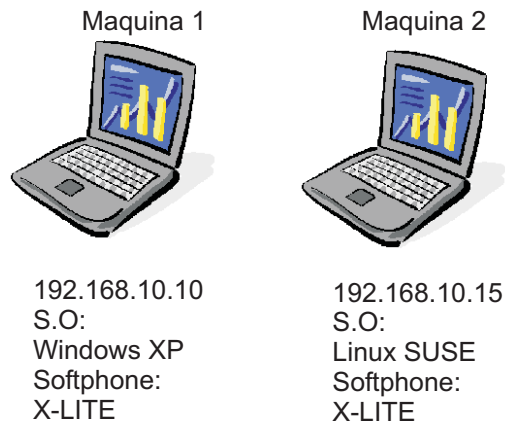


Figura 5.1: Especificaciones técnicas de los clientes

5.7.2. Configuración de Asterisk SIP.conf

Para la configuración local se necesitarán entonces dos canales de comunicación de tipo SIP y para definir dichos canales se debe editar el fichero *sip.conf* [35] que se localiza en el directorio */etc/asterisk* de SUSE.

Al igual que en la mayoría de los lenguajes de comunicación, (como Lenguaje C, Visual Basic, entre otros), se pueden incluir comentarios en el editaje precedidos por el signo *;* indicando de esta manera la función de cada línea. Entonces se procede a incluir la definición de los canales en el fichero *sip.conf* quedando como siguen:

[general]

context=default

srvlookup=yes

[110]

type=friend

secret=mexico2

context=internalcalls

regexten=110

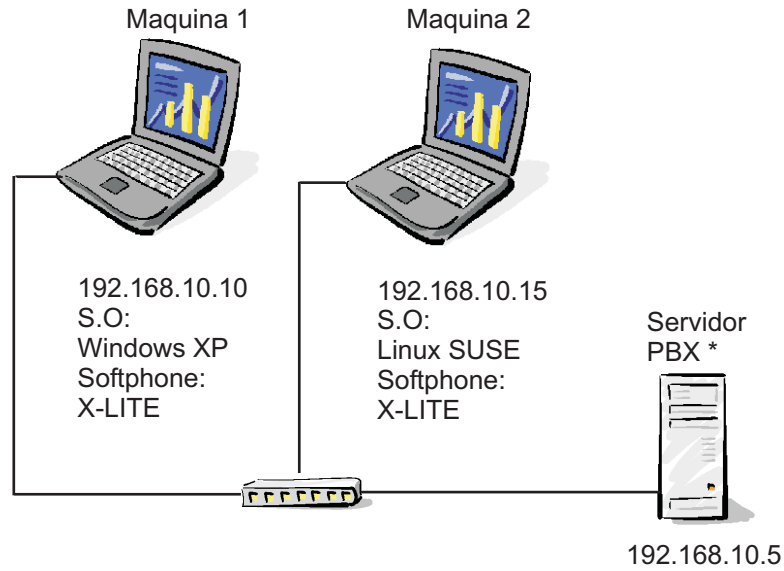


Figura 5.2: Interconexion de clientes SIP Y PBX asterisk en una red local

callerid=tele

qualify=yes

nat=no

host=dynamic

canreinvite=no

mailbox=110@firstmail

progressinband=yes

[115]

type=friend

secret=mexico3

context=internalcalls

regexten=115

callerid=xmen

qualify=yes

nat=no

host=dynamic

canreinvite=no

mailbox=115@firstmail

progressinband=yes

La Tabla 5.2 muestra la funcionalidad de cada parámetro definido para cada uno de los canales.

PARAMETRO	FUNCIONALIDAD
[general]	Se encuentra al inicio del archivo sip.conf y contiene las configuraciones generales de los canales y sus opciones.
context=default	Contexto predefinido para todo tipo de usuario (pertenecce al contexto [general])
srvlookup=yes	Activa la opción que permite remitir las llamadas a diferentes ubicaciones sin necesidad de cambiar su dirección lógica; puede ubicar las llamadas basadas en nombres de dominio (pertenecce al contexto [general])
[ejemplo]	Contiene el nombre de la extensión o canal a configurar
type=friend	Especifica el tipo de extensión, si realiza llamadas es tipo peer, si solo recibe llamadas es tipo user y si realiza y recibe es de tipo friend
secret=welcome	Es la contraseña de autenticación
context=nombre_de_contexto o	Es quizás de los parámetros más importantes pues especifica la localización de las instrucciones de marcado usadas para controlar las llamadas entrantes y salientes de esa extensión; este nombre de contexto busca correspondencia con el nombre de contexto especificado en extensions.conf
regexten=num_de_extensión	Genera la extensión especificada para ese canal
callerid=nombre_mostrado	Es el nombre que se desplegará en los métodos de funcionamiento y solicitudes de SIP
qualify=yes	Habilita la opción de monitoreo de latencia del canal; la latencia debe ser menor a 2ms (2segundos)
nat=no	Si la extensión se encuentra detrás de un NAT debe habilitarse con un yes y si no como en este caso solo no se habilita
host=dynamic	Estableciéndolo como dinámico permite utilizar la extensión para registrarnos y la dirección IP se conocerá al realizarse el registro, si en cambio se quiere limitarlo a una sola IP se cambia el dynamic por la dir. IP.
canreinvite=no	Se utiliza para redireccionar llamadas, estableciendo que no se le permita a Asterisk ser el intermediario y no permitir envío de mensajes directamente entre clientes
mailbox=extensión@miservidor	Es la dirección de correo de voz en la cual se guardaran estos mensajes para poder ser escuchados por el usuario, para este se configura de manera anexa el archivo voicemail.conf
progressinband=yes	Permite escuchar los tonos de progreso, ocupado o de colgar de una llamada.

Tabla 5.2: Parámetros establecidos en la definición para SIP.CONF

5.7.3. Configuración de Asterisk extension.conf

El dialplan es, sin duda alguna, el corazón de cualquier servidor SIP, puesto que define de que manera debe manejar Asterisk las llamadas entrantes y salientes de los clientes. El dialplan consiste, en una serie de instrucciones o pasos de marcado que Asterisk debe seguir rigurosamente y que se configuran en el fichero *extension.conf* [35]. Está compuesto básicamente de cuatro partes principales: los contextos, extensiones, prioridades y aplicaciones.

A) Contextos

Los contextos son grupos de extensiones. En ellos se establecen diferentes partes del dialplan para interactuar con algún otro contexto o con los canales definidos. Es denotado colocando el nombre de contexto entre corchetes []. Y todo lo que después de el se coloque pertenecerá a si mismo hasta que no se defina otro contexto.

B) Extensiones

En cada contexto se define una o más extensiones. Una extensión es una instrucción que el PBX de Asterisk seguirá activándose una llamada entrante o al marcar dígitos en un canal. En su sentido común las extensiones también son usadas para especificar números de extensiones telefónica. La sintaxis para una extensión es la palabra *exten*, seguida por una flecha formada por el signo igual y el signo mayor que, como se muestra abajo:

exten =>

Una extensión consta de tres componentes principales:

- El nombre o número de la extensión.
 - La prioridad (cada extensión puede incluir múltiples pasos, el número de paso es llamado prioridad).
 - La aplicación o comando que ejecutará alguna acción durante la llamada.
-

El nombre, prioridad y aplicación son tres componentes que se separan por comas como se muestra a continuación:

exten => nombre, prioridad, aplicación()

C) Prioridades

Cada paso a seguir en una extensión es llamada prioridad. Cada prioridad es numerada secuencialmente, iniciando con 1. Cada prioridad ejecuta una aplicación en particular. Por ejemplo, la siguiente extensión responderá el teléfono (como prioridad 1) y después colgará (como prioridad 2):

exten => 123,1,Answer ()

exten => 123,2,Hangup ()

D) Aplicaciones

Cada aplicación ejecuta una acción específica en el canal que utilicen en ese momento, tales como reproducir un sonido, aceptar un tono de entrada, o colgar una llamada. En el ejemplo anterior se muestran dos aplicaciones muy simples: Answer() y Hangup ().

Una vez que se ha conocido la estructura de configuración de un dialplan y se ha analizado la configuración de *sip.conf* es posible observar que los dos canales para esta implementación están asociados al mismo contexto de nombre [internalcalls]. Por lo tanto en el archivo *extensions.conf*, que se encuentra en el directorio */etc/asterisk/*, se debe en primera instancia definir el contexto [internalcalls] como un dialplan.

[internalcalls]

exten => 110,1,Dial(SIP/110,10,r)

exten => 110,2,Voicemail(u110@firstmail)

exten => 110,102,Voicemail(b110@firstmail)

```
exten => 110,103,Hangup

exten => 115,1,Dial(SIP/115,10,r)

exten => 115,2,Voicemail(u115@firstmail)

exten => 115,102,Voicemail(b115@firstmail)

exten => 115,103,Hangup

exten => 150,1,Voicemailmain
```

En el dialplan [internalcalls] se están creando las extensiones asociadas a los clientes (esto es extensión 110 y extensión 115) y se establecen como disponibles dentro del contexto [internalcalls]. La aplicación Dial(), como prioridad 1, crea un canal tipo SIP con los "peers" de nombre 110 y 115.

La aplicación Voicemail(uxten@miservidor) envía al correo de voz cuando la extensión que se marca no se encuentra disponible o el cliente no contesta y Voicemail(bexten@miservidor) envía al correo de voz cuando la extensión que se marca se encuentra ocupada. La aplicación Hangup() termina la llamada. Y la aplicación Voicemailmain, asignada a la extensión 150, envía al usuario al menú principal del correo de voz.

En el segundo dialplan se establece que el comportamiento ideal para las llamadas entrantes desde las extensiones o el PBX Asterisk es el siguiente: una vez que una llamada entra por la línea analógica, se accede a un sistema automático de atención u operadora. El sistema (el PBX Asterisk) preguntará por la extensión deseada, una vez que el número de extensión se introduzca por la terminal (tonos DTMF) la llamada se encaminará a una de las extensiones.

La idea principal es permitir compartir un sólo número del PBX con todas las extensiones. Esta configuración se implementa en el fichero *extensions.conf*. Por lo tanto se agrega un nuevo contexto denominado [incomingcalls] cuya programación queda como sigue:

```
[incomingcalls]

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

```
exten => s,2,DigitTimeout(10)

exten => s,3,ResponseTimeout(20)

exten => s,4,Background(vm-extensions)

exten => i,1,Goto(incomingcalls,s,1)

exten => t,1,Hangup()

include => internalcall
```

El funcionamiento del dialplan [incomingcalls] es el siguiente: Se contesta la llamada y en seguida se configura el valor máximo para introducir el número de la extensión. En la siguiente aplicación la voz de la operadora pide el número de la extensión; si el número de la extensión no existe la aplicación *Goto* envía al inicio del contexto para preguntar de nueva cuenta.

Al final se establece la aplicación para terminar la llamada y la opción *include* permite enlazar los dos contextos y por consecuencia los dos dialplans configurados.

5.7.4. Configuración de Asterisk voicemail.conf

El siguiente paso para la estructuración del servidor SIP en la red local es configurar el archivo *voicemail.conf* [35], ubicado en el directorio */etc/asterisk/* de SUSE, para habilitar el correo de voz que se proporciona para las extensiones en el dialplan. El archivo *voicemail.conf* también maneja contextos en donde se pueden especificar diferentes correos de voz. La sintaxis para la definición de un correo de voz es la siguiente:

```
mailbox => contraseña,nombre,[email]
```

En donde cada uno de los valores a configurar determina lo siguiente:

- **mailbox:** Es el nombre de correo de voz. Regularmente corresponde con el número de extensión de la configuración asociada.
-

- contraseña: Es la contraseña numérica con la cual el dueño de la extensión accede a su correo de voz. Si el usuario cambia la contraseña, el sistema actualiza este valor en el archivo *voicemail.conf*.
- nombre: Se coloca el nombre del dueño del correo. Se puede usar el callerid.
- mail: Se establece la dirección de correo electrónico del dueño de la extensión. En este correo se recibirán las notificaciones e incluso el mismo correo de voz que el PBX Asterisk procesará.

De tal manera que la configuración del archivo *voicemail.conf* quedaría como sigue:

```
[firstmail]
```

```
110 => 5678,Tele,tele@localhost
```

```
115 => 9012,Xmen,Xmen@localhost
```

5.7.5. Configuración de los clientes

La configuración de los clientes se realiza de manera directa. Las partes más importantes son el nombre de usuario y la contraseña para el registro; además de la dirección del servidor Asterisk con el cual se van a registrar.

La primera parte de la configuración de los clientes es la instalación del software telefónico softphone, que para la presente implementación es el X-LITE de X-TEN que puede descargarse del sitio web <http://www.xten.com>.

Una vez descargado e instalado el softphone en los dos clientes SIP se procede a la configuración. La Figura 5.3 muestra un ejemplo de configuración de un cliente usando este softphone. Los parámetros a configurar en cada uno de los clientes son:

- Display name:

Se coloca el nombre que se haya especificado en la sección callerid del archivo *sip.conf* para cada uno de los clientes.

- User name:

Es un nombre utilizado para identificar a la extensión, por lo regular se coloca el nombre especificado como nombre de la extensión para cada cliente en el archivo *sip.conf*.

- Authorization user:

También utilizado para la autenticación de los clientes frente al servidor SIP; se coloca también el nombre de la extensión especificado para cada cliente en el archivo *sip.conf*.

- Domain/realm:

Es uno de los parámetros más importantes de la configuración del cliente; aquí se establece la dirección IP del servidor SIP.

- SIP Proxy:

En el parámetro SIP Proxy se coloca la dirección IP del servidor SIP pero seguido del número de puerto utilizado para la comunicación de SIP (es decir X.X.X.X:5060).

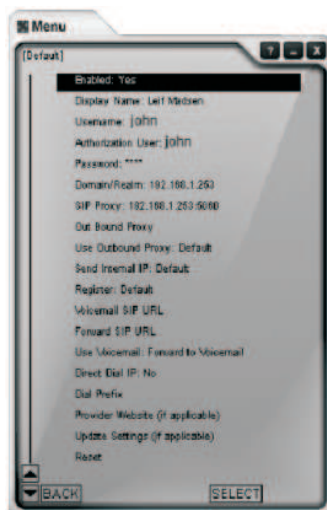


Figura 5.3: Pantalla de configuracion softphone X-LITE

De tal modo que la configuración de cada uno de los clientes se muestra en la Tabla 5.3.

CLIENTE MAQUINA #1	S.O. WINDOWS XP
Display name	tele
User name	110
Authorization user	110
Domain/Realm	192.168.10.5
SIP Proxy	192.168.10.5:5060
CLIENTE MAQUINA #2	S.O. LINUX SUSE
Display name	xmen
User name	115
Authorization user	115
Domain/Realm	192.168.10.5
SIP Proxy	192.168.10.5:5060

Tabla 5.3: Parámetros configurados en cada uno de los clientes SIP

5.7.6. Conectividad entre clientes Windows, Linux y Servidor SIP

El primer paso para realizar las pruebas de conectividad y operatividad entre cada uno de los clientes SIP, que como se mencionó se encuentran en dos plataformas completamente diferentes, y el servidor SIP es inicializar al servidor SIP, que preferentemente se debe arrancar en modo de depuración. Esto se lleva a cabo insertando el siguiente comando en la terminal de linux:

```
/usr/sbin/asterisk -cvvv
```

Con lo cual se ingresa al funcionamiento del PBX Asterisk y a la CLI del mismo. El siguiente punto es registrar a los clientes con el servidor SIP a través de sus respectivos softphones, esto es, ingresar a las aplicaciones que activan a las terminales telefónicas tanto en Windows XP como en Linux SUSE. Es importante mencionar que las imágenes de los softphones se obtuvieron directamente del software X-Lite del fabricante X-Ten. Entonces en la CLI se obtienen las siguientes líneas:

```
Asterisk Ready.
```

**CLI>*

– Registered SIP '110' at 192.168.10.10 port 55674 expires 3600

– Registered SIP '115' at 192.168.10.15 port 5060 expires 1800

Y en los clientes SIP se puede observar su registro como se muestra en las Figura 5.4



Figura 5.4: Registro de cliente en un servidor sip

Una vez que los clientes se han registrado de manera exitosa en el servidor SIP se procede a realizar las pruebas de llamadas. La Figura 5.5 muestra la forma en la cual el cliente MAQUINA 1, con extensión 110, realiza una llamada hacia el cliente MAQUINA 2, con extensión 115. El resultado se muestra en la Figura 5.6 en donde se observa el establecimiento de la llamada.



Figura 5.5: Solicitud de llamada de ext 110 a ext 115



Figura 5.6: Llamada entre máquina 1 y máquina 2 establecida

Una vez que se ha establecido la llamada cualquiera de los dos clientes SIP puede terminarla para iniciar una nueva o tener la línea disponible. Cabe mencionar que el softphone de Linux tiene la capacidad de establecer hasta tres llamadas simultaneas pues cuenta con tres líneas disponibles, mientras que el softphone de Windows con dos líneas disponibles permite dos llamadas simultáneas. La Figura 5.7 muestra el momento en el que cualquiera de los dos clientes a finalizado la llamada.



Figura 5.7: Terminacion de llamada entre máquina 1 y máquina 2

El proceso de establecimiento, mantenimiento y finalización de la llamada anterior entre los dos clientes SIP mediante la gestión del servidor SIP arroja las siguientes líneas en la CLI del PBX Asterisk:

**CLI>*

- Executing Dial("SIP/110-2ce4", "SIP/115/10|r") in new stack*
- Called 115*
- SIP/115-9632 is ringing*
- SIP/115-9632 answered SIP/110-2ce4*
- User Hangup*

Las siguientes Figuras muestran el mismo proceso de llamada pero ahora del cliente MAQUINA 2 (Ext.115) hacia el cliente MAQUINA 1 (Ext.110). Para esta llamada de la ext.115 a la ext.110 se obtienen las siguientes líneas de monitoreo en la CLI del servidor:

**CLI>*

- Executing Dial("SIP/115-9d6d", "SIP/110/10|r") in new stack*
- Called 110*
- SIP/110-3b80 is ringing*
- SIP/110-3b80 answered SIP/115-9d6d*
- User Hangup*

Se configuraron ambas líneas para que tuvieran correo de voz y se pudiera ingresar a el, mediante la configuración del fichero *voicemail.conf*, de tal manera que si cualquiera de los dos teléfonos marca a la otra extensión y el usuario no se encuentra en ese momento disponible para contestar, el teléfono automáticamente envía al cliente que esta realizando la llamada hacia el correo de voz para dejar un mensaje.

A continuación se muestran las líneas de monitoreo de la CLI del PBX Asterisk en las cuales se puede observar el proceso de acceso al correo de voz.



Figura 5.8: Solicitud de llamada máquina 2 a máquina 1

- En primer término se tiene una llamada de la ext.110 (MAQUINA 1) hacia la ext.115 (MAQUINA 2) y el servidor SIP envía al usuario de la MAQUINA 1 al correo de voz que se definió para la MAQUINA 2.

**CLI>*

- *Executing Dial("SIP/110-2a65", "SIP/115/10|r") in new stack*
- *Called 115*
- *SIP/115-b9de is ringing*
- *Nobody picked up in 10000 ms*
- *Executing VoiceMail("SIP/110-2a65", ü115@firstmail") in new stack*
- *Playing 'vm-theperson'(language én')*
- *Playing 'vm-theperson'(language én')*
- *Playing 'digits/1'(language én')*
- *Playing 'digits/1'(language én')*
- *Playing 'digits/5'(language én')*



Figura 5.9: Llamada entre máquina 2 y máquina 1 establecida



Figura 5.10: Llamada entre máquina 2 y máquina 1 terminada

– *Playing 'vm-isunavail'(language én')*

– *Playing 'beep'(language én')*

– *Recording the message*

– *x=0, open writing:*

/var/spool/asterisk/voicemail/firstmail/115/INBOX/msg0000

format: wav49,0x814b440

– *x=1, open writing:*

/var/spool/asterisk/voicemail/firstmail/115/INBOX/msg0000

format: gsm, 0x814b7d0

- x=2, open writing:

/var/spool/asterisk/voicemail/firstmail/115/INBOX/msg0000

format: wav, 0x80f7b58

- User hung up

- Las siguientes líneas de monitoreo corresponden a una llamada de la ext.115 (MAQUINA 2) hacia la ext.110 (MAQUINA 1) y el servidor SIP envía al usuario de la MAQUINA 2 al correo de voz que se definió para la MAQUINA 2.

**CLI>*

- Executing Dial("SIP/115-2a91", "SIP/110/10|r") in new stack

- Called 110

- SIP/110-3d21 is ringing

- Nobody picked up in 10000 ms

- Executing VoiceMail("SIP/115-2a91", ü110@firstmail") in new stack

- Playing 'vm-theperson'(language én')

- Playing 'vm-theperson'(language én')

- Playing 'digits/1'(language én')

- Playing 'digits/1'(language én')

- Playing 'digits/0'(language én')

- Playing 'vm-isunavail'(language én')

- Playing 'beep'(language én')

– *Recording the message*

– *x=0, open writing:*

/var/spool/asterisk/voicemail/firstmail/110/INBOX/msg0000

format: wav49,0x814b7d0

– *x=1, open writing:*

/var/spool/asterisk/voicemail/firstmail/110/INBOX/msg0000

format: gsm, 0x80f7de8

– *x=2, open writing:*

/var/spool/asterisk/voicemail/firstmail/110/INBOX/msg0000

format: wav, 0x80f7ef8

– *User hung up*

De igual forma el correo de voz esta configurado para iniciarse si cualquiera de los usuarios con los que se quisiera establecer una llamada se encuentra ocupado con otra llamada (esto implica que todas sus líneas disponibles están ocupadas, si no la llamada se direcciona a cualquiera de sus líneas disponibles). A continuación se muestran las líneas de monitoreo de la CLI del PBX Asterisk en las cuales se puede observar este proceso.

- Las siguientes líneas de monitoreo corresponden a una llamada de la ext.115 (MAQUINA 2) hacia la ext.110 (MAQUINA 1) y el servidor SIP detecta que ambas líneas del usuario MAQUINA 1, (que como se mencionó anteriormente en plataforma Windows cada usuario tiene dos líneas disponibles a comparación de Linux en donde cada usuario tiene tres líneas disponibles), están ocupadas y envía al usuario MAQUINA 2 al correo de voz por tono de ocupado que se definió para el usuario MAQUINA 1.

**CLI>*

- *Executing Dial("SIP/115-d8d9", "SIP/110/10|r") in new stack*
- *Called 110*
- *SIP/110-abc5 is ringing*
- *SIP/110-abc5 answered SIP/115-d8d9*
- *Attempting native bridge of SIP/115-d8d9 and SIP/110-abc5*
- *Started music on hold, class 'default', on SIP/115-d8d9*
- *Started music on hold, class 'default', on SIP/110-abc5*
- *Executing Dial("SIP/115-39a0", "SIP/110/10|r") in new stack*
- *Called 110*
- ***Got SIP response 486 "Busy Here"back from 192.168.10.10***
- ***SIP/110-241a is busy***
- ***== Everyone is busy/congested at this time***
- ***Executing VoiceMail("SIP/115-39a0", "b110@firstmail") in new stack***
- *Playing 'vm-theperson'(language én')*
- *Playing 'digits/1'(language én')*
- *Playing 'digits/1'(language én')*
- *Playing 'digits/0'(language én')*
- ***Playing 'vm-isonphone'(language én')***
- *Playing 'vm-intro'(language én')*
- *Playing 'beep'(language én')*
- *Recording the message*
- *x=0, open writing:*

/var/spool/asterisk/voicemail/firstmail/110/INBOX/msg0001 format:

wav49, 0x814b1c0

– *x=1, open writing:*

/var/spool/asterisk/voicemail/firstmail/110/INBOX/msg0001 format: gsm,

0x80f8620

– *x=2, open writing:*

/var/spool/asterisk/voicemail/firstmail/110/INBOX/msg0001 format:

wav,0x81989d8

– *User hung up*

Una vez que se ha accedido al correo de voz de cualquiera de los dos clientes SIP y que se ha registrado y grabado un mensaje de voz, el softphone está configurado para indicar que tiene un mensaje de voz nuevo en su bandeja de entrada, esto lo hace desplegando en su pantalla un ícono de un sobre de correo que es la imagen representativa de un correo nuevo.

La Figura 5.11 muestra el ícono de correo de voz nuevo desplegado en las pantallas del softphone para cada uno de los clientes SIP. Para poder acceder al menú principal del correo de voz de cada uno de los clientes SIP, se creó una extensión más en el dialplan en el fichero *extensions.conf*. Esta extensión es la número 150, por tanto, para poder acceder al menú principal del correo de voz de cada uno de los clientes SIP se marca esta extensión como se muestra en la Figura 5.12.

Una vez que se ha entrado al menú principal una operadora solicitará de manera automática que se registre al correo, esto es, ingresar el nombre de usuario que en este caso es el número de extensión del cliente que quiera acceder al correo de voz y después la contraseña que fue preestablecida al configurar el fichero *voicemail.conf*. Lo restante es seguir las instrucciones de la operadora para escuchar los correos de voz o acceder a las configuraciones del mismo correo.

Es importante recordar que como todo correo de voz, el que se definió para los usuarios, tiene entre su menú una opción para cambiar la contraseña de acceso al correo de voz,

cuando se modifica la contraseña por el usuario esta se modificará automáticamente en el fichero voicemail.conf sin necesidad de acceder a este o sin que se presenten conflictos.



Figura 5.11: Softphone mostrando que hay nuevo correo de voz

A continuación se muestran las líneas de monitoreo desplegadas en la CLI del PBX Asterisk para el acceso al menú principal del correo de voz realizada por los clientes SIP.



Figura 5.12: Clientes estableciendo llamada con el correo de voz

- Acceso del cliente MAQUINA 1 a su correo de voz (el cliente MAQUINA 1 escucha uno de sus correos):

**CLI>*

- *Executing VoiceMailMain("SIP/110-b451",) in new stack*
 - *Playing 'vm-login'(language én')*
 - *Playing 'vm-password'(language én')*
 - *Playing 'vm-youhave'(language én')*
 - *Playing 'digits/2'(language én')*
 - *Playing 'vm-INBOX'(language én')*
 - *Playing 'vm-and'(language én')*
 - *Playing 'digits/1'(language én')*
 - *Playing 'vm-Old'(language én')*
 - *Playing 'vm-message'(language én')*
 - *Playing 'vm-onefor'(language én')*
 - *Playing 'vm-INBOX'(language én')*
 - *Playing 'vm-messages'(language én')*
 - *Playing 'vm-opts'(language én')*
 - *Playing 'vm-helpexit'(language én')*
 - *Playing 'vm-first'(language én')*
 - *Playing 'vm-message'(language én')*
 - *== Parsing '/var/spool/asterisk/voicemail/firstmail/110/INBOX/msg0000.txt':Found*
 - *Playing 'vm-received'(language én')*
 - *Playing 'digits/at'(language én')*
 - *Playing 'digits/5'(language én')*
 - *Playing 'digits/30'(language én')*
 - *Playing 'digits/3'(language én')*
-

- *Playing 'digits/a-m'(language én')*
- *Playing '/var/spool/asterisk/voicemail/firstmail/110/INBOX/msg0000'(language én')*
- *Playing 'vm-advopts'(language én')*
- *Playing 'vm-repeat'(language én')*
- *Playing 'vm-next'(language én')*
- *Playing 'vm-delete'(language én')*
- *Playing 'vm-toforward'(language én')*
- *Playing 'vm-savemessage'(language én')*
- *Playing 'vm-helpexit'(language én')*
- *Playing 'vm-goodbye'(language én')*

- Acceso del cliente MAQUINA 2 a su correo de voz (solo accesa):

**CLI>*

- *Executing VoiceMailMain("SIP/115-fc45",) in new stack*
 - *Playing 'vm-login'(language én')*
 - *Playing 'vm-password'(language én')*
 - *Playing 'vm-youhave'(language én')*
 - *Playing 'digits/2'(language én')*
 - *Playing 'vm-INBOX'(language én')*
 - *Playing 'vm-and'(language én')*
 - *Playing 'digits/2'(language én')*
 - *Playing 'vm-Old'(language én')*
 - *Playing 'vm-messages'(language én')*
-

- *Playing 'vm-onefor'(language én')*
- *Playing 'vm-INBOX'(language én')*
- *Playing 'vm-messages'(language én')*
- *Playing 'vm-opts'(language én')*
- *Playing 'vm-helpexit'(language én')*
- *Playing 'vm-goodbye'(language én')*

Con esto queda comprobada la operatividad de las configuraciones en los ficheros *sip.conf*, *extensions.conf* y *voicemail.conf*, además se demostró la correcta funcionalidad e interoperabilidad de los clientes SIP con el servidor SIP y entre clientes a nivel de una red local.

5.8. Utilidad y ventajas del protocolo SIP

5.8.1. Convergencia e interoperabilidad del protocolo SIP

La convergencia e interoperabilidad de las redes telefónicas y las redes de datos es una de las tendencias tecnológicas más importantes de nuestra época. El potencial de esta unión es símbolo de un gran avance e impulso, siendo capaz de originar notables mejoras y ahorros en las redes de telecomunicaciones [36].

El objetivo es ofrecer soluciones que aprovechen la infraestructura de red IP, con el propósito de mejorar la efectividad y productividad de los servicios. Hasta hace algún tiempo, la mayoría de los sectores usuarios de las telecomunicaciones poseían una PBX para la conmutación pública en la red telefónica y una red LAN completamente independiente para el transporte de datos

En los últimos años se han ido haciendo cada vez más populares los sistemas CTI (integración telefónica computacional) que relacionan las redes de voz y de datos. La convergencia e interoperabilidad de la infraestructura telefónica y de datos permite simplificar la administración de los recursos de red y facilita la expansión de capacidades.

La ventaja real de la unión datos-telefonía es la opción que plantea para soportar nuevas aplicaciones para el usuario.

El impulso tecnológico que está haciendo posible la integración de las redes de voz y de datos es el crecimiento y la difusión de las redes IP, tanto a nivel LAN como a nivel WAN. En los siguientes diez años, la conectividad IP alcanzará un grado de inserción similar a la utilización de electricidad en el hogar o la empresa [36]. Algunas de las ventajas fundamentales que ofrece la tecnología de redes IP y que repercuten en los servicios telefónicos son:

1. Las redes IP desaparecen los límites físicos asociados a los teléfonos y funcionalidades telefónicas tradicionales. Dentro de poco será posible acceder simultáneamente a todos los servicios tradicionales y a la capacidad de responder llamadas desde cualquier lugar del mundo, sin que la parte originadora dependa de su posición geográfica. Así se permite ofrecer un servicio flexible para viajeros frecuentes y sitios remotos.
2. El protocolo IP es independiente de la capa de enlace, permitiendo que los usuarios finales elijan el formato de enlace más adecuado. IP puede viajar sobre ATM, ethernet, frame relay, ISDN o incluso mediante líneas analógicas.
3. Un conjunto de estándares universales relacionados a las redes IP permitirá a muchos proveedores ofrecer productos compatibles. Cada uno de los estándares harán posible la competencia entre múltiples fuentes de servicios de red y hardware. La competencia minimizará los costos y maximizará los nuevos servicios para el usuario final.
4. La convergencia de las redes de datos y las redes telefónicas es decisiva para la evolución de la industria de servidores SIP que utilizan PBXs SIP. La tendencia más importante prevista será la migración desde una estructura predominantemente compuesta por sistemas de conmutación pública a una estructura más abierta y con sistemas compatibles sobre el formato IP.

La estructura de comunicaciones de servidores SIP sobre redes IP incluye cuatro grandes áreas de desarrollo [36]:

- Infraestructura IP: Básicamente compuesta por la conectividad IP provista principalmente por los proveedores de equipamiento LAN/WAN.
- Control de llamada (sistemas operativos y servidores): Sistemas operativos LAN con la capacidad de proveer servicios y servidores SIP basados en PBXs SIP para las funcionalidades telefónicas tradicionales.
- Dispositivos de usuario: Software (softphones) y teléfonos IP, capaces de ser conectados a redes IP directamente con niveles de calidad similares a la red telefónica tradicional.
- Aplicaciones avanzadas: Aprovechando la natural integración de los sistemas telefónicos y de datos, han surgido y surgirán aplicaciones muy sofisticadas que proporcionan los servicios telefónicos clásicos tales como IVR (respuesta de voz interactiva) y call centers (centro de llamadas).

Es importante tener en cuenta que la calidad y confiabilidad de la arquitectura PBX SIP bajo la infraestructura de red IP son aspectos muy importantes en la introducción de esta nueva tecnología, ya que dichos atributos deben ser comparables con los niveles de la red telefónica pública.

Algunas de las funcionalidades incluidas en los servidores SIP que utilizan PBXs SIP son:

- Resistencia a cortes de abastecimiento de energía.
- Configuración redundante en el servidor para aumentar la confiabilidad en el control de llamada.
- Enrutamiento de llamada alternativo cuando los enlaces IP o los enlaces telefónicos no están disponibles.

5.8.2. Nivel de utilidad del protocolo SIP

A partir de 1998 se logro que los proveedores de equipos y servicios pudiesen concentrarse en desarrollar aplicaciones de valor agregado que se necesitan para llevar la

demanda de la telefonía IP en base a servidores SIP al ser la telefonía IP una alternativa de bajo costo ante los servicios tradicionales de larga distancia que ofrecen los grandes monopolios de telefonía [36].

Con todo esto la telefonía IP empieza a ser una realidad cotidiana en muchas empresas por la rápida compensación y el ahorro de costos que proporciona. Aunque hasta ahora básicamente son sólo las empresas las que adoptan el cambio por la reducción de costos que esto implica.

Como ejemplo se puede decir, que si una empresa coloca un servidor SIP la inversión que esto representa se recupera antes de un año con las llamadas de carácter local, y en tres meses en las empresas que llaman tres horas al día con su oficina situada en América o Asia.

Existen una serie de ventajas que impulsan la introducción de la tecnología de la que se desprende el protocolo SIP. Estas ventajas tienen diversos orígenes y van evolucionando de acuerdo a los cambios del mercado. Entre las más importantes se puede mencionar sin duda alguna el ahorro de larga distancia al realizar llamadas telefónicas sobre infraestructura de red IP o a través de Internet.

La reducción de costos de capital es otra de las ventajas que conllevan los servidores SIP, así como los costos de administración. Un ejemplo de la simplicidad de operación ganada con las redes IP integradas es la conexión de un teléfono al sistema. Sólo es necesario conectarlo a la red para que inmediatamente alcance el estado operacional normal.

Otra ventaja es la utilización de canales de banda ancha (ADSL/XDSL y módem de cable) que permiten una calidad total en cuanto a las llamadas respecta, además en consecuencia, la introducción en el mercado del hogar de estas nuevas tecnologías permitirá el desarrollo de multimedia IP.

Capítulo 6

Conclusiones

Los conmutadores telefónicos, también llamados PBX (Private Branch eXchange o, en español, central telefónica para servicio privado), suelen hacer todos lo mismo: recibir n líneas telefónicas desde el exterior de una empresa, y enlazarlas con m extensiones, dentro de la misma. Por supuesto, m suele ser un número mayor que n , y en el sencillo detalle de poder manejar un número muy grande de usuarios radica la principal ventaja de contar con uno equipo PBX.

De manera que, se llega a la conclusión de que los PBXs facilitan y optimizan el uso de extensiones y líneas telefónicas, motivo por el cual estas centrales telefónicas conforman la estructura básica y fundamental del sistema de telefonía pública tradicional

Tanto la telefonía IP como VoIP presentan una estructura de comunicación telefónica totalmente basada en redes IP que son sin duda alguna la opción de mayor utilidad para la integración de sistemas de comunicaciones.

Ambos sistemas de comunicación (tanto telefonía IP como VoIP) ofrecen la posibilidad de trabajar bajo la estructura del SIP, que con el paso del tiempo será el protocolo mas funcional para la convergencia de voz y datos.

Por lo tanto, se concluye que la tecnología VoIP está desarrollada bajo una infraestructura que permite el mayor aprovechamiento posible de los recursos con los que se cuenten en la red de datos, y, solamente incorpora algunos dispositivos que son de lo mas accesibles y versátiles.

Por supuesto que la mayor ventaja que ofrecen las tecnologías sobre IP es el bajo costo que implica su implementación y el gran ahorro que representan con respecto a la telefonía tradicional y proveedores telefónicos con su alto número de equipos

Es importante concluir que una de las mayores ventajas que ofrece SIP es la sencillez de sus estructura y la facilidad con que puede ser utilizado comparándose en calidad de servicio con la PSTN tradicional.

Muchos puntos terminales de SIP solamente necesitan soportar la base de especificaciones SIP y tal vez unas cuantas extensiones de control para proporcionar una extensa variedad de usos y servicios innovadores

Se determinó a través de este trabajo de tesis la estructura básica de un mensaje SIP y también se pudo distinguir entre los múltiples tipos de respuesta que se presentan al momento de establecer, mantener y finalizar una sesión y que son de una gran importancia pues ofrecen la alternativa de ser una guía de resolución de problemas al momento de estructurar un sistema en base a este protocolo.

En conclusión se puede recurrir a todas las clases de respuestas para conocer en que se esta fallando, es decir, se pueden ocupar como guía para la solución de errores.

SIP es un protocolo que se relaciona en funcionamiento con los otros protocolos que son de utilidad general en nuestros tiempos, tales como RTP, TCP, UDP, SDP, entre otros, los cuales hacen aún más fácil el uso de SIP al momento de implementar un servidor de telefonía IP.

Además, el SIP es el protocolo que mayor concentración de funcionalidades tiene y esto ha hecho que a comparación de otros protocolos, SIP, sea el que mayor auge esté teniendo y para el cual se estén desarrollando más y más aplicaciones compatibles para usuarios finales.

La tecnología de telefonía IP utilizando el protocolo SIP crece continuamente, tanto en número de usuarios como en inversiones realizadas por las empresas. Esta tecnología ha ido evolucionando año con año hasta lo que es ahora, una completa realidad, aunque todavía le queda un largo camino por recorrer.

Es importante mencionar que el protocolo SIP puede ser utilizado bajo cualquier platafor-

ma de sistema operativo, proporcionando la misma calidad de llamada que en la red de telefonía pública. La telefonía IP usando el protocolo SIP permite que los usuarios con un equipo de cómputo pueda conectarse a la central telefónica de SIP y la permita realizar llamadas desde cualquier lugar en el que se encuentre ubicado.

Gracias a la aplicación desarrollada en esta tesis se puede concluir que el simple hecho de poder converger la tecnología de voz y datos hace posible el aprovechar al máximo las capacidades de nuestra red. Es posible, por lo tanto, diseñar una completa red de telefonía con extensiones y salida a la PSTN en base a la estructura de datos con la que ya se cuenta en una empresa, una escuela, un edificio, en el hogar.

Apéndice A

Acrónimos

AAA Authentication, Authorization y Accounting, Funciones de autenticación y cobro

ACF Mensaje de Confirmación de Admisión

ACK Acknowledgement, Confirmación de recepción

ADSL Asymmetric Digital Subscriber Line, Línea de Abonado Digital Asimétrica

AIB Authenticated Identity Body, Campo de autenticación de información

ANSI American National Standards Institute, Instituto Nacional Estadounidense de Estándares

API Application Programming Interface, Interfaz de Programación de Aplicaciones

ARJ Mensaje de Rechazo de Admisión

ARQ Mensaje de Petición de Admisión

Asterisk Es una aplicación de código abierto de una central telefónica (PBX)

ATM Asynchronous Transfer Mode, Modo de Transferencia Asíncrona

Backbone (La parte de una red que actúa como ruta primaria para el tronco que sale y llega de otras redes con mayor frecuencia).

Bayonne (Servidor de telefonía del GNU).

BHCA Busy Hour Call Attempts, Intento de Marcado en Horas Congestionadas

BRJ Bandwidth Reject, Rechazo de Banda ancha

BRQ Bandwidth Request, Solicitud de Banda ancha

Call-Id Identificador de Llamada

Carrier Class Término utilizado como distintivo de gran calidad

CBX Computerized Branch Exchange, Central Telefónica Computarizada

CDR Call Detail Records, Registro Detallado de Llamada

CLEC Competitive Local Exchange Carrier, Portador Local Competitivo de Intercambio

CLI Command Line Interface, Interfaz de Línea de Comandos

CONTACT Contiene una (o más) dirección que pueden ser usada para contactar con el usuario

CPIM Common Presence and Instant Messages, Presencia común y mensajería instantánea

CRC Cyclical Redundancy Codes, Códigos de Redundancia Cíclica o códigos polinómicos

Cseq Secuencia de comandos. Se inicia con un número aleatorio e identifica de forma secuencial cada petición

DATE La cabecera de campo DATE (fecha) se usa para transportar la fecha de cuando se envió una solicitud o una respuesta

DCF Confirmación de liberación

DHCP Dynamic Host Configuration Protocol, Protocolo de Configuración Dinámica de Nodos

DNS Domain Name Server, Servidor de Nombres de Dominio

DRQ Solicitud de Liberación

DTMF Dual-tone multi-frequency, Tono doble de multifrecuencia

E1 Enlace de 2 megabits por segundo de ancho de banda

E3 Enlace de 32 megabits por segundo de ancho de banda

Ethernet Una red física y protocolo de enlace que opera en las dos primeras capas del modelo de referencia OSI a velocidades de 10, 100 o 1000 Mbps

ETSI European Telecommunications Standards Institute, Instituto Europeo de Estándares de Telecomunicaciones

Frame Relay Frame-mode Bearer Service, Técnica de comunicación mediante retransmisión de tramas

FROM De. Indica la dirección del origen de la petición

FTP File Transfer Protocol, Protocolo de transferencia de archivos.

FXO Foreign Exchange Office, Dispositivo de computador que permite conectarse a la PSTN

FXS Foreign Exchange Station, Sirven para conectar teléfonos analógicos normales a un computador

Gatekeeper GK. Es un servidor opcional que es responsable de los servicios de la red incluyendo el registro, admisión y estado de las terminales, para lo cual se usa un protocolo especial llamado RAS

Gateway GW. Es una entidad en la red que actúa como puente hacia otras redes.

GnomeMeeting Es una aplicación de software libre para realizar videoconferencias y telefonía por IP en base a la recomendación H.323

GNU/Linux Es la denominación defendida por Richard Stallman y otros para el sistema operativo que utiliza el kernel Linux en conjunto con las aplicaciones de sistema creadas por el proyecto GNU y de varios otros proyectos/grupos de software

GSM Global System for Mobile Communications, Sistema Global para Comunicaciones Móviles

H.323 Es una recomendación del ITU-T que define los protocolos para proveer sesiones de comunicación audiovisual sobre paquetes de red

HTTP HyperText Transfer Protocol, Protocolo de Transferencia de Hipertexto

IANA Internet Assigned Numbers Authority, Agencia de Asignación de Números de Internet

IETF Internet Engineering Task Force, Fuerza de Tarea de Ingeniería en Internet

IM Instant Messages, Mensajes Instantáneos

IN Intelligent Network, Red Inteligente

INVITE El método de solicitud INVITE es utilizado para establecer sesiones multimedia entre agentes de usuarios

IP Internet Protocol, Protocolo de Internet

IPv4 Internet Protocol Version 4

IPv6 Internet Protocol Version 6

ISDN Integrated Digital Services Network, Red Digital de Servicios Integrados (RDSI)

ISPs Internet Service Providers, Proveedores de Servicios de Internet

ISUP Es un protocolo de circuitos conmutados, usado para configurar, manejar y gestionar llamadas de voz y datos sobre PSTN

ITU International Telecommunication Union, Unión Internacional de Telecomunicaciones

LAN Local Area Network, Red de Area Local

LDAP Lightweight Directory Access Protocol, Protocolo para Acceso a un Directorio Básico

MCU Multicast Control Unit, Unidad de Control Multipunto

MGC Media Gateway Control Protocol, Control del Gateway de Medios diseñado por la IETF).

MGCP Media Gateway Control Protocol, Protocolo de Control del Gateway de Medios / Controlador de Gateway de Medios

MIME Multipurpose Internet Mail Extensions, Extensiones de Correo Internet Multipropósito

MMUSIC Multiparty Multimedia Session Control, Control de Sesión Multimedia Multipartida

NAT Network Address Translation, Traducción de Direcciones de Red

NetMeeting Aplicación gratuita de Microsoft que permite conversar con usuarios a través de audio, video y texto

OSI Open System Interconnection, Interconexión de Sistemas Abiertos

PABX Private Automatic Branch Exchange, Central Telefónica para Servicio Privado Automatizado

PAX Private Automatic Exchange, Servicio Privado Automatizado

PBX Private Branch Exchange, Central Telefónica para Servicio Privado

PC Personal Computer, Computadora Personal

PCI Peripheral Component Interconnect, Interconexión de Componentes Periféricos

PCM Pulse Code Modulation, Modulación por Pulsos Codificados

peer to peer En inglés peer-to-peer -que se traduciría de par a par o de punto a punto, y más conocida como P2P , se refiere a una red que no tiene clientes ni servidores fijos, sino una serie de nodos que se comportan simultáneamente como clientes y como servidores de los demás nodos de la red

PPP Point-to-Point Protocol, Protocolo punto a punto

PRI Primary Rate Interface, Interfaz de Acceso Primario

Proxy Hace referencia a un programa o dispositivo que realiza una acción en representación de otro

PSTN Public Switched Telephone Network, Red Telefónica Pública Conmutada

PTTs Autoridades de correo, teléfonos y telégrafos).

QoS Quality of Service, Calidad de Servicio

R2 Es un protocolo peer to peer, es decir, no hay cliente y servidor, los dos dispositivos se comunican con simple señalización de 4 bits

RADIUS Servicio para Acceso Remoto del Usuario por Línea Conmutada

RAS Registration, Admission and Status, Registro, Admisión y Estado

REGISTER El método de solicitud REGISTER es utilizado por un agente de usuario para notificar a una red de trabajo SIP su actual recurso de identificación uniforme de contacto CONTACT-URI (dirección IP)

RFCs Request for Comments, Estándares oficiales de los protocolos de internet

Router Enrutador, encaminador. Dispositivo de hardware o software para interconexión de redes de computadoras que opera en la capa tres del modelo OSI

RSVP Resources V Protocol, Protocolo de Reserva de Recursos

RTP Real-time Transport Protocol, Protocolo de Transporte en Tiempo real

SDP Session Description Protocol, Protocolo de Descripción de Sesión

SIP Session Initiation Protocol, Protocolo de Inicialización de Sesión

SMTP Simple Mail Transfer Protocol, Protocolo Simple de Transferencia de Correo electrónico

SONET Synchronous Optical Networking, Red de Sincronización Óptica, es un estándar para el transporte de telecomunicaciones en redes de fibra óptica

SS7 Servicio de señalización 7

SSL Secure Sockets Layer, Capa de Seguridad de Conexión

SUBJECT La cabecera de campo opcional SUBJECT (asunto) se usa para indicar el asunto de la sesión de los medios de comunicación

SYN Sincronización.

T1 Sistema portador con tasa de transmisión de 1,544 Mbps

T3 Sistema portador con tasa de transmisión de 44,736 Mbps

TCAP Transaction Capabilities Application Part, Parte del uso de las Capacidades de la Transacción

TCP Transmission Control Protocol, Protocolo para Control de Transmisión

TDM Time-division multiplexing, Multiplexación por División de Tiempo).

TLS Transport Layer Security, Seguridad para Capa de Transporte

TO Para. Indica la dirección del destinatario de la petición

UA User Agent, Agente de Usuario

UAC User Agent Client, Agente de Usuario Cliente

UAS User Agent Server, Agente de Usuario Servidor

UDP User Datagram Protocol, Protocolo de Usuario de Datagrama

URI Uniform Resource Identifier, Identificador Uniforme de Recursos

URL Uniform Resource Locator, Localizador Uniforme de Recurso

User Agent Contiene el cliente agente que realiza la comunica

VIA Indica el transporte usado para el envío e identifica la ruta del request, por ello cada proxy añade una línea a este campo

VoIP Voice over IP, Voz sobre Protocolo de Internet

VPN Virtual Private Network, Red Privada Virtual

WAN Wide Area Network, Red de Área Amplia

WEB World Wide Web

Bibliografía

- [1] Trabajos universitarios UTEM (Universidad Tecnológica Metropolitana de Santiago de Chile)“*Surgimiento de los PBX e inicios de la telefonía privada*”,<http://html.starmedia.com/informatica/pbx.html> 2006
- [2] Soluciones integrales telefonía total “*Que es y cuales son las características principales de un PBX*”,<http://www.telefoniatotal.com/serviciosysoluciones/telefoniatradicional.htm> 2007
- [3] Revista RED “*Generalidades de los sistemas PABX, PCX y CBX*”,<http://www.red.com.mx/index.php> glosario abril 2007
- [4] pabxcontrol “*Principales tipos de sistemas que pueden ser confundidos con un PBX*”,<http://www.pabxcontrol.com/index.php> 2007
- [5] Cisco Systems, Inc“*Generalidades de los PBX modernas*”,<http://www.cisco.com/global/ES/solutions/.shtml> 2006
- [6] MPSnet“*funciones especiales de los PBX*”<http://www.mpsnet.com.mx/index.php> 2006
- [7] “*La telefonía IP en la actualidad 10*”,<http://www.monografias.com/trabajos10/tele/tele.shtml>
- [8] “*voz sobre IP y telefonía tradicional*”,<http://www.comtest.com/tutorials/VoIP.html>
- [9] Voice over IP (VoIP)“*característica principales de la voz sobre IP*”,<http://www.monografias.com/trabajos3/voip/voip.shtml> 2000
- [10] EL ESTANDAR VOIP - VOZ SOBRE IP “*La telefonía IP en el mundo*”,<http://www.enredes.com.ar/antivirus/voip.htm> 2003

-
- [11] Jose Antonio Sanchez Ortiz, Ingeniero de Telecomunicaciones, Telefonía IP “*Generalidades de la telefonía IP*”, <http://www.monografias.com/especiales/telefonaiip/index.shtml>
 - [12] Handley, M., “*SIP: Session Initiation Protocol*”, RFC 2543, 1999.
 - [13] Rosenberg, J., “*SIP: Session Initiation Protocol*”, RFC 3261, 2002.
 - [14] Bradner, S., “*The Internet Standards Process: Revision 3*”, RFC 2026, 1996.
 - [15] Leiner, B., “*A Brief History of the Internet*”, <http://www.isoc.org/internet/history/brief.html>. The Internet Society.
 - [16] “*Internet Protocol*”, RFC 791, 1981.
 - [17] Deering, S., y R. Hinden, “*Internet Protocol, Version 6 (IPv6) Specification*”, RFC 1883, 1995.
 - [18] “*Transmission Control Protocol*”, RFC 793, 1981.
 - [19] Postal, J., “*User Datagram Protocol*”, RFC 768, 1980.
 - [20] Dieks, T., “*The TLS Protocol Version 1.0*”, RFC 2246, 1999.
 - [21] Stewart, R., “*Stream Control Transmission Protocol*”, RFC 2960, 1999.
 - [22] Sparks, R., “*The Session Initiation Protocol (SIP) Refer Method*”, RFC 3515, 2003.
 - [23] Roach, A., “*Session Initiation Protocol (SIP)-Specific Event Notification*”, RFC 3265, 2002.
 - [24] Campbell, B., “*Session Initiation Protocol (SIP) Extension for Instant Messaging*”, RFC 3428, 2002.
 - [25] Crocker, D., “*Common Presence and Instant Messaging (CPIM)*”, IETF Internet-Draft, en progreso, 2003.
 - [26] Donovan, S., “*The SIP INFO Method*”, RFC 2976, 2000.
 - [27] Donovan, S., y J. Rosenberg, “*Session Timers in the Session Initiation Protocol (SIP)*”, IETF Internet-Draft, en progreso, 2003.
-

-
- [28] Peterson, J., “*Enhancements for Authenticated Identity Management in the Session Initiation Protocol (SIP)*”, IETF Internet-Draft, en progreso, 2003.
 - [29] Peterson, J., “*SIP Authenticated Identity Body (AIB) Format*”, IETF Internet-Draft, en progreso, 2003.
 - [30] “*Packet-Based Multimedia Communications Systems*”, ITU Recommendation H.323, 2000.
 - [31] “*Linux*”, <http://es.wikipedia.org/wiki/Linux>.
 - [32] “*Linux: Etimologías*”, <http://es.wikipedia.org/wiki/Linux/Etimolog.C3.ADa>.
 - [33] “*Distribuciones Linux*”, <http://es.wikipedia.org/wiki/Linux/DistribucionesLinux>.
 - [34] “*What is Asterisk*”, <http://www.asterisk.org/about>.
 - [35] Van Meggelen, Jim, J. Smith, y M. Leif, “*Asterisk: The Future of the Telephony*”, <http://safari.oreilly.com/0596009623>, O'Reilly Media, 2005.
 - [36] Wallingford, Theodore, “*Switching to VoIP*”, <http://safari.oreilly.com/0596008686>, O'Reilly Media, 2005.
-