

ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL
Facultad de Ingeniería en Electricidad y Computación

INFORME DE MATERIA DE GRADUACIÓN

**“Instalación y Configuración del Chan Mobile en
Asterisk y su Conexión a la Red GSM”**

Previa a la obtención del Título de:

INGENIERO EN TELEMÁTICA

INGENIERO EN TELEMÁTICA

**INGENIERO EN COMPUTACIÓN ESPECIALIZACIÓN SISTEMAS
TECNOLÓGICOS**

Presentada por:

WENDY ELIZABETH ANDRADE IZURIETA

JASON ISRAEL JIJÓN ALBÁN

FREDY LEONARDO BUENAÑO VINUEZA

GUAYAQUIL – ECUADOR

AÑO

2009

AGRADECIMIENTO

A Dios, nuestro Señor, nuestro Guía, nuestro Proveedor, por lo esencial que ha sido en nuestra posición firme de alcanzar esta meta.

A nuestras familias, por darnos la estabilidad emocional, económica y sentimental para poder llegar hasta este logro el cual, definitivamente, no hubiese podido ser realidad sin ustedes.

A nuestros profesores, que con paciencia y sabiduría nos impartieron conocimiento y experiencia, ingrediente primordial para nuestro progreso intelectual en el largo camino del desarrollo mental y espiritual.

DEDICATORIA

A Dios, guía y luz de mi camino. A mis padres, hermanos y profesores por ser la fuente de mis valores y conocimientos.

Wendy E. Andrade Izurieta

A mi madre, tías, abuela y hermano por su amor y apoyo incondicional. Hacen de mí un mejor ser humano cada día.

Jason I. Jijón Albán

Agradezco a Dios, a mis padres y a mis compañeros que estuvieron junto a mí en este largo camino a cumplir un sueño.

Freddy L. Buenaño Vinuesa

DECLARACIÓN EXPRESA

“La responsabilidad del contenido de esta Tesis de Grado, nos corresponde exclusivamente; y el patrimonio intelectual de la misma a la Escuela Superior Politécnica del Litoral”.

(Reglamento de Graduación de la ESPOL)

Wendy Elizabeth Andrade Izurieta

Jason Israel Jijón Albán

Freddy Leonardo Buenaño Vinuesa

TRIBUNAL DE GRADUACIÓN

Ing. Rebeca Estrada Pico

PROFESOR DIRECTOR DE LA MATERIA DE GRADUACIÓN

Ing. Patricia Chávez

PROFESOR DELEGADO POR EL DECANO DE LA FACULTAD

RESUMEN

El proyecto a realizar consiste en la implementación de una centralita telefónica basada en Asterisk capaz de acceder a la red GSM por medio del módulo Chan Mobile. El mismo que permite usar celulares como gateways a través de un enlace bluetooth, para el flujo de llamadas salientes desde la red de telefonía IP conectada al servidor Asterisk que requieran establecer comunicación con dispositivos móviles.

Con la implementación de este proyecto buscamos administrar de forma eficiente los recursos económicos como tecnológicos, para su uso empresarial considerando los siguientes puntos:

- a) El costo de adquisición del hardware a utilizar.
- b) Las llamadas entre operadores celulares siempre son más baratas.
- c) El control sobre la duración de las llamadas.
- d) Jerarquía de acceso a los canales de salida entre los usuarios.
- e) Uso de mensajes de texto como alternativa adicional a las llamadas debido a que son más baratos y su impacto sobre el servidor es menor que la realización de llamadas.

INTRODUCCION

En el cambiante mundo de las telecomunicaciones, la masificación del uso de dispositivos móviles, de los cuales destaca el celular, ha hecho posible la comunicación personalizada en diferentes ámbitos cotidianos. Es por eso que la capacidad de conexión con estos dispositivos se hace necesaria en las centrales telefónicas, especialmente en las empresariales.

Sin embargo, un gran impedimento era el costo económico tanto del servicio en sí como del hardware a utilizar para su implementación. No obstante en la actualidad, debido en gran parte a la competencia y regulaciones estatales, los precios de las llamadas celulares han bajado considerablemente en una misma operadora, aunque se mantiene relativamente igual cuando se realiza una interconexión entre operadoras de telefonía celular.

La central telefónica Open Source Asterisk, mediante el módulo Chan Mobile, proporciona un método eficaz para combatir los problemas de costos de implementación a la vez que permite una distribución del flujo saliente de llamadas entre operadoras y un flexible control del mismo.

INDICE DE CONTENIDO

CAPITULO I: ANTECEDENTES Y JUSTIFICACION.....	1
1.1 ANTECEDENTES	2
1.2 JUSTIFICACION	3
1.3 DESCRIPCION DEL PROYECTO	3
1.3.1 Objetivos Generales.....	3
1.3.2 Objetivos Específicos	4
1.4 METODOLOGIA	7
1.5 PERFIL DE LA TESIS	7
CAPITULO II: ASTERISK Y LIBRERÍA CHAN MOBILE	9
2.1 ASTERISK	10
2.1.1 Funcionalidades de Asterisk	13
2.1.2 Compatibilidad.....	15
2.2 FXS Y FXO	16
2.1.3 FXS	17
2.1.4 FXO.....	17
2.3 CHAN MOBILE	18
2.3.1 Historia de Chan Mobile	18
2.3.2 Chan Mobile Trabajando con Asterisk.....	19
2.3.3 Características.....	20
CAPITULO III: IMPLEMENTACION.....	22
3.1 INTRODUCCION	23
3.2 HARDWARE	23
3.2.1 Servidor.....	24
3.2.2 Bluetooth	24
3.2.3 Gateways	25
3.2.4 Teléfonos IP	26
3.3 SOFTWARE	27

3.3.1	Servidor PBX.....	27
3.3.2	BlueZ	28
3.3.3	Softphones.....	28
3.4	INSTALACIÓN.....	29
3.4.1	Instalación de Librerías Base	29
3.4.2	Instalación de Librerías para Bluetooth.....	30
3.4.3	Instalación de Asterisk.....	30
3.5	CONFIGURACION DE ARCHIVOS DE ASTERISK	32
3.5.1	Configuración SIP.CONF.....	32
3.5.2	Configuración EXTENSIONS.CONF	36
3.5.3	Configuración MOBILE.CONF.....	57
CAPITULO IV: FUNCIONAMIENTO Y PRUEBAS.....		60
4.1	INICIALIZANDO E INGRESANDO A ASTERISK.....	61
4.2	CONFIGURANDO EXTENSIONES SIP.....	62
4.2.1	Configurando Softphone XTEN-XLITE	63
4.2.2	Configuración GRANDSTREAM	64
4.3	VINCULACION DE DISPOSITIVOS MOVILES	67
4.3.1	Dirección MAC del Dispositivo Bluetooth Dongle	67
4.3.2	Dirección MAC y Puerto de Dispositivos Móviles.....	68
4.4	LLAMADAS A TELEFONOS CELULARES.....	70
4.4.1	Llamadas a Porta.....	70
4.4.2	Llamadas a Movistar	72
4.4.3	Llamadas a Alegro.....	73
4.5	ENVIO DE MENSAJES DE TEXTO A TELEFONOS CELULARES.....	74
4.5.1	Envío de Mensaje Plantilla 1 (Ver Tabla VI)	75
4.5.2	Envío de Mensaje Plantilla 2 (Ver Tabla VII)	76
4.5.3	Envío de Mensaje Plantilla 3 (Ver Tabla VIII)	77
4.5.4	Envío de Mensaje Plantilla 4 (Ver Tabla IX)	78
4.5.5	Envío de Mensaje Plantilla 5 (Ver Tabla X)	79
CONCLUSIONES Y RECOMENDACIONES		80

TRABAJO A FUTURO	¡Error! Marcador no definido.
SOLUCION AL PROBLEMA DE PORTABILIDAD	¡Error! Marcador no definido.
GLOSARIO DE TERMINOS.....	83
A N E X O S	86
ANEXO A	87
PORTABILIDAD NUMERICA	87
IMPLEMENTACION DE PORTABILIDAD NUMERICA	87
FUNCIONAMIENTO DE ALL CALL QUERY	88
BILIOGRAFIA	89

INDICE DE FIGURAS

Fig. 1.1 Esquema de la Implementación	5
Fig. 2.1 Software Asterisk para central telefónica	10
Fig. 2.2 Conexión de Asterisk	11
Fig. 2.3 Interfaz en una aplicación móvil	14
Fig. 2.4 Interfaz de Asterisk Win32PBX	15
Fig. 2.5 FXS/FXO SIN PBX	17
Fig. 3.2 Teléfono GXP2000	26
Fig. 3.1 Teléfono GXP1200	26
Fig. 3.3 Diagrama de flujo para envío de SMS	42
Fig. 3.4 Diagrama de macro para conectarse a RED GSM	53
Fig. 4.1 Configuración Xten-Xlite	63
Fig. 4.2 Administración via web - login Grandstream	64
Fig. 4.3 Administración vía web - Configuración Grandstream, account 1 normal	65
Fig. 4.4 Administración vía web-Configuración Grandstream, account 2 privilegiada	66
Fig. 4.5 Administración vía web-Reboot Grandstream	66
Fig. 4.6 Obtención dirección Mac dispositivo Bluetooth dongle	67
Fig. 4.7 Obtención de dirección MAC dispositivo móvil	68
Fig. 4.8 Obtención del puerto de dispositivo móvil	68
Fig. 4.9 Petición de vinculación del Servidor asterisk al dispositivo móvil	69
Fig. 4.10 Consola remota de Asterisk – vinculación exitosa	69
Fig. 4.11 Consola remota de Asterisk – Listado de Dispositivos acoplados	70
Fig. 4.12 Consola remota de Asterisk – Llamada a Porta	71
Fig. 4.13 Dispositivos móviles – Llamada a Porta	71
Fig. 4.14 Consola remota de Asterisk – Llamada a Movistar	72
Fig. 4.15 Dispositivos móviles – Llamada a Movistar	73
Fig. 4.16 Consola remota de Asterisk – Llamada a Alegro	73
Fig. 4.17 Dispositivos móviles – Llamada a Alegro	74
Fig. 4.18 Dispositivos móviles – Envío de mensaje plantilla 1 (ver tabla VI)	75
Fig. 4.19 Dispositivos móviles – Envío de mensaje plantilla 2 (ver tabla VII)	76
Fig. 4.20 Dispositivos móviles – Envío de mensaje plantilla 3 (ver tabla VIII)	77
Fig. 4.21 Dispositivos móviles – Envío de mensaje plantilla 4 (ver tabla IX)	78
Fig. 4.22 Dispositivos móviles – Envío de mensaje plantilla 5 (ver tabla X)	79

INDICE DE TABLAS

Tabla I Características del Servidor	24
Tabla II Bluetooth Dongle	25
Tabla III Gateways	25
Tabla IV Servidor PBX	27
Tabla V Plantilla de Mensaje	42
Tabla VI Plantilla de Mensaje 1	44
Tabla VII Plantilla de Mensaje 2	45
Tabla VIII Plantilla de Mensaje 3	46
Tabla IX Plantilla de Mensaje 4	47
Tabla X Plantilla de Mensaje 5	48
Tabla XI Series Numéricas de celulares Alegro	49
Tabla XII Series Numéricas de Celulares Movistar	50
Tabla XIII Series Numéricas de Celulares Porta	51

CAPITULO 1

ANTECEDENTES Y JUSTIFICACION

1.1 ANTECEDENTES

En el mundo empresarial actual, las comunicaciones juegan un papel esencial y pueden ser decisivas en el éxito o fracaso de un modelo de negocios. Es por esto que la elección de un sistema de comunicación que se ajuste a las necesidades de una empresa en particular debe ser tratada de manera especial por el empresario.

En años recientes ha emergido un software para PBX que alcanzado un gran nivel de popularidad en la comunidad mundial debido en gran medida a su flexibilidad además de todos los beneficios de ser Open Source, su nombre es Asterisk. Este programa ha permitido ahorrar en costosas centrales telefónicas privativas que obligan a la dependencia de una marca ya que en gran medida no cumplen con estándares reconocidos.

Uno de los problemas que resuelve Asterisk es el manejo de canales de salida para telefonía de distinta naturaleza, entre las cuales se encuentra la red GSM, usada ampliamente para la comunicación entre dispositivos móviles. Tradicionalmente la conexión a la red GSM es a través de una tarjeta adicional FXS/FXO en el servidor Asterisk y una base celular. Sin embargo, gracias al concepto de Open Source, la comunidad de desarrolladores ha contribuido con un modulo que nos permite evitar la compra de dichos dispositivos y usar celulares convencionales que soporten conexión bluetooth, así como también proveer al servidor PBX de un enlace de este tipo.

1.2 JUSTIFICACION

Debido a la gran demanda de recursos económicos necesarios para implementar una central telefónica en un ambiente empresarial en un esquema privativo, surge la necesidad de buscar una alternativa más económica y de mayor flexibilidad que se ajuste a las necesidades específicas del cliente. Una de estas alternativas es el uso de Software libre para las implementaciones de redes de voz usando el estándar de comunicación VoIP.

Sin embargo los canales de salida comúnmente utilizados son los que pertenecen a las compañías de telefonía local, por lo tanto cuando se requieren hacer llamadas a teléfonos celulares los costos son mayores en comparación que si se las realizaran entre líneas pertenecientes a la misma operadora celular.

1.3 DESCRIPCION DEL PROYECTO

La implementación de nuestra solución tecnológica pretende alcanzar los siguientes objetivos.

1.3.1 Objetivos Generales

- Implementar un sistema PBX que permita la comunicación con la red GSM, de manera sencilla, rápida, escalable y de bajo costo, dentro de un ambiente

empresarial, a fin de cubrir las continuas exigencias propias del crecimiento y desarrollo de las redes de voz.

1.3.2 Objetivos Específicos

- Independizar el flujo de llamadas a teléfonos móviles del resto de posibles tipos de tráfico telefónico.
- Promover el uso de redes tipo PAN (Personal Area Network) para tráfico de VoIP.
- Definir un plan eficaz para el manejo de concurrencia de llamadas que hagan un uso correcto de los recursos disponibles.
- Aislar el tráfico telefónico hacia la red GSM separándolo por operadoras de telefonía celular.
- Promover la implementación de software libre en centrales telefónicas para su uso empresarial como reemplazo a centrales privativas.

El proyecto a realizar consiste en la implementación de una centralita telefónica basada en Asterisk capaz de acceder a la red GSM por medio del módulo Chan Mobile, el mismo que permite usar celulares como gateways a través de un enlace bluetooth, para el flujo de llamadas salientes desde la red de telefonía IP conectada al servidor Asterisk que requieran establecer comunicación con dispositivos móviles.

El esquema a utilizar en este proyecto se define en la figura 1.1. El número de celulares usados como gateways será igual o mayor a la cantidad de operadoras de telefonía móvil, habiendo por lo menos un celular de una determinada compañía proveedora de servicios de telefonía móvil. El acceso a los gateways de salida se hará tomando en consideración la operadora celular del número de destino de la llamada; es decir, las llamadas salientes se asignan a un gateway según su operadora para que las llamadas se hagan dentro de la misma red de la empresa telefónica correspondiente.

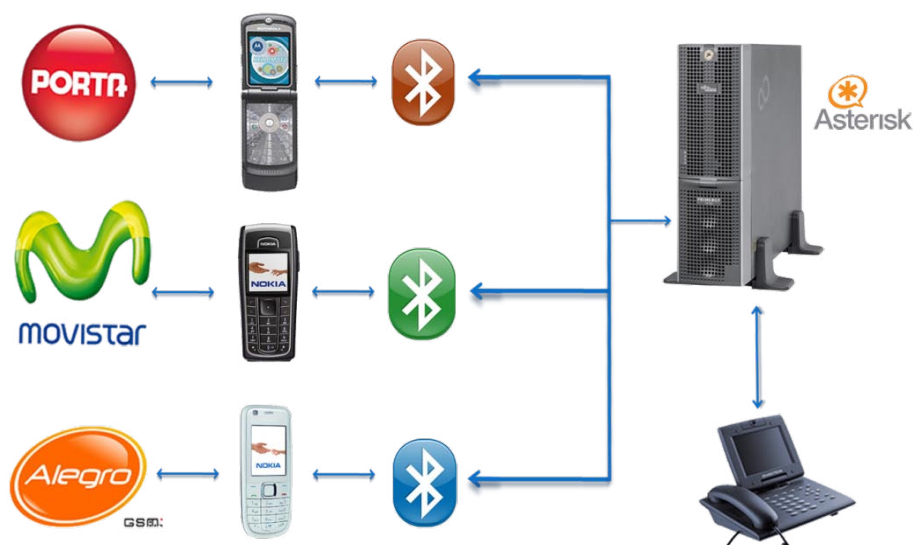


Fig. 1.1 Esquema de la Implementación

Nuestra central manejará colas de espera para llamadas concurrentes cuando no haya disponible ningún gateway de la operadora celular del destino. Sin embargo, cuando la llamada se origine de un rango de extensiones en particular, nombradas de aquí en adelante como Extensiones

Privilegiadas, las cuales serán previamente definidas y que pertenecen a la red de telefonía IP conectada al servidor Asterisk, la elección del dispositivo para la salida de la llamada a la red GSM se hará según la disponibilidad de todos los celulares usados como gateways sin importar la operadora de telefonía a la que pertenecen. Si todos los dispositivos están siendo ocupados, entonces las llamadas de estas extensiones privilegiadas se enviarán igualmente a un estado de espera, el cual es el mismo para todas las extensiones.

La duración de las llamadas a la red GSM será limitada para las extensiones normales, pero las pertenecientes a las extensiones privilegiadas tendrán duración ilimitada. Además se limitará el número de intentos de acceso a un dispositivo de salida en la cola de espera de llamadas.

De esta manera se realizará una administración eficiente de los gateways de acceso a la red GSM, debido a que éstos son recursos limitados que deben ser correctamente gobernados. La implementación permite la escalabilidad, ya que en cualquier momento se podrá adicionar un gateway de una determinada compañía celular para incrementar la disponibilidad y la capacidad de llamadas concurrentes que el sistema pueda soportar.

1.4 METODOLOGIA

Para cumplir nuestros objetivos realizaremos una instalación de Asterisk sobre un servidor con sistema operativo Red Hat Enterprise Linux. Dotaremos al servidor de conexiones bluetooth para poder establecer los canales de comunicación con operadoras de telefonía celular. Se efectuará una configuración el manejo de llamadas salientes y el envío de mensajes de texto predefinidos.

1.5 PERFIL DE LA TESIS

Nuestra tesis tiene como objetivo principal establecer y optimizar el acceso a la red GSM desde una central VoIP de Asterisk haciendo uso de la librería Chan Mobile.

En el capítulo 2, se revisa los fundamentos teóricos, para así comprender el salto tecnológico seguido con VoIP, sus características, mecanismos de implementación, aplicaciones y servicios que proporciona actualmente, además de las proyecciones que tiene la tecnología para su futuro.

En el capítulo 3 se detallara las especificaciones técnicas de la solución, el análisis, diseño y la implementación del proyecto.

Posteriormente, en el capítulo 4, se realizara las pruebas de conexión, el establecimiento de las llamadas, la concurrencia, el encolamiento, y la escalabilidad.

CAPITULO 2

ASTERISK Y LIBRERÍA CHAN MOBILE

2.1 ASTERISK



Fig. 2.1 Software Asterisk para central telefónica

Asterisk es un software tipo PBX (Private Branch Exchange y Private Automatic Branch Exchange para PABX), lo que significa que funciona como una central secundaria privada automática, puedes obtener a partir de su empleo una central telefónica conectada directamente a la red pública de teléfono por medio de líneas troncales para gestionar, además de las llamadas internas, las entrantes y/o salientes con autonomía sobre cualquier otra central telefónica. Está diseñado originalmente para que funcione con Linux, pero trabaja muy bien con BSD, Windows (emulado) y OS X. Asterix funciona a partir del protocolo IP y puede interfuncionar con casi todo el equipo de telefonía basado en los estándares usando un hardware relativamente económico. Provee servicios voicemail (correo de voz), comunicación directa, identificación de llamadas,

respuesta de voz interactiva y llamada en espera. Para ello emplea servicio de llamadas ID con los protocolos SIP, H323, ADSI y IAX. Para funcionar con voz sobre IP no necesita de ningún hardware adicional, ahora para interconectar con la telefonía tradicional requiere de tarjetas especiales que se instalan en el computador y que son de muy bajo costo como las conocidas tarjetas FXO y FXS.



Fig. 2.2 Conexión de Asterisk

Es bueno señalar que el empleo de sistemas PBX evita conectar todos los teléfonos de una oficina de manera separada a la red de telefonía local pública (RTC), ya que funciona como un switch de red, con ello te ahorras el empleo de una línea propia con salidas de llamadas y cargos mensuales hacia la central telefónica que regresan nuevamente para establecer comunicación interna. Si

posees una oficina pequeña y deseas emplear software PBX tradicional, tus costos serían altos, pero con Asterisk se reducen los costos de instalación ya que emula este tipo de centrales con lo que puedes aprovechar las funciones de este tipo de sistemas.

La solución Open Source de telefonía basada en el software Asterisk desarrollado por Digium, Inc, te provee de un vasto conjunto de funciones. Asterisk ofrece las funciones propias de las centralitas clásicas y además características avanzadas, pudiendo trabajar tanto con sistemas de telefonía estándar tradicionales como con sistemas de Voz sobre IP. A pesar de su distribución libre y gratuita Asterisk está dotado con características que sólo puedes encontrar en grandes sistemas PBX de alto costo, así posees las funcionalidades antes descritas e inclusive buzón de voz, conferencia por medio de voz y registros de llamada detallados.

Para la interconexión con el equipo de telefonía digital y análogo, Asterisk se apoya en varios dispositivos de hardware, estos tiene la opción de interconexión simple o cuádruple T1 e interfaces E1 para la interconexión a líneas PRI y bancos de canal. Sin embargo como señalamos puedes emplear puertos cuádruples FXO o puertos FXS, los que están disponibles y son populares para pequeñas instalaciones; estas tarjetas pueden ser empleadas para cuatro salidas BRI (ISDN2) o puertos de ocho salidas BRI compatibles con tarjetas HFC. Para la interconexión con la red celular (GSM o CDMA), Asterisk puede usar al conductor de canal Celliax (el cual funciona con Asterix versión 1.2) o Chan

Mobile para lo cual se dispone de una versión no oficial en Voip-info.org.

2.1.1 Funcionalidades de Asterisk

- Sistema de menú en Pantalla ADSI (Interfaz Analógico para presentación de Servicios).
- Receptor de alarmas Agregar Mensaje (Append Message).
- Autenticación de Llamadas con respuesta automatizada.
- Opciones de transferencia de llamadas no supervisada (automatizada).
- Opciones de registros de llamada detallados.
- Desvío de llamadas al interno en el caso que la extensión está ocupada o no responde.
- Desvío de llamada variable.
- Monitorización de llamadas, con opciones de aparcamiento de llamadas.
- Sistema de grabación de llamadas.
- Recuperación de llamadas (DID y ANI).
- Sistema de escucha de llamadas.
- Opciones de transferencia de llamadas.
- Configuración de llamada en espera.
- Identificación de llamadas con opciones de bloqueo, este sistema también se aplica a las llamadas en espera.

- Configurable para trabajar con conferencia de voz.
- Almacenamiento y recuperación en base de datos.
- Integración con Base de Datos.
- Opciones de marcado por nombre.
- Acceso directo al sistema interno.
- Tonos de llamada distintivos.
- Empleo de agentes locales y remotos.
- Configuración de música a tu elección para el proceso de espera, con un sistema de reproducción aleatoria y control de volumen.
- Marcación predictiva.
- Opciones de privacidad.
- Protocolo de establecimiento abierto (Open Settlement Protocol – OSP).

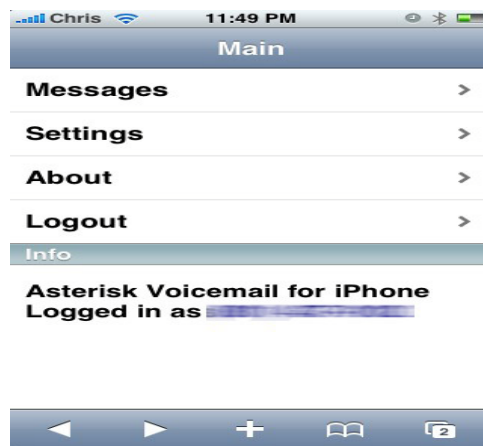
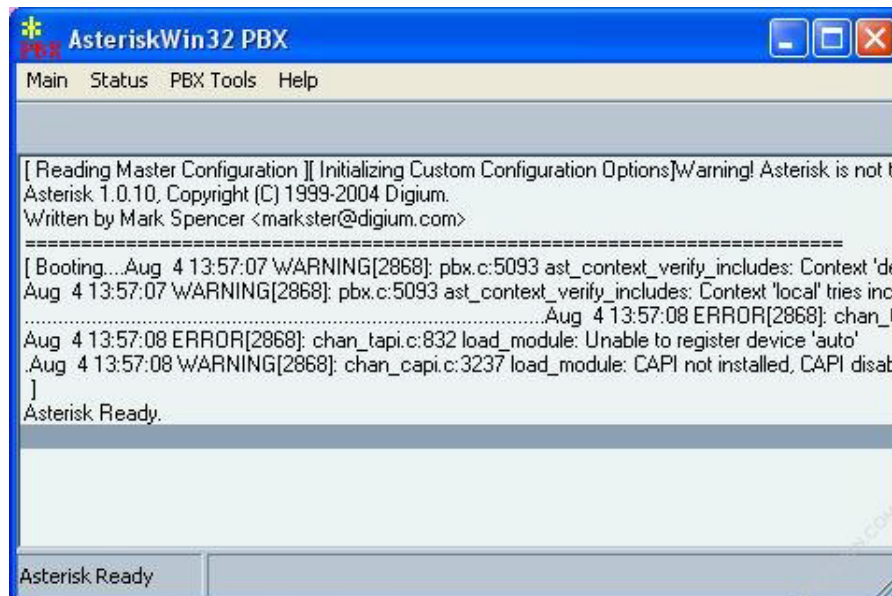


Fig. 2.3 Interfaz en una aplicación móvil

|

2.1.2 Compatibilidad

Asterisk funciona en muchas plataformas OS, siendo Linux la principal plataforma para el desarrollo y soporte del hardware desarrollado por Digium, por lo que es más conveniente instalar Asterisk PBX en Debian Gnu/Linux. Sin embargo si para nuestro uso es suficiente el empleo de VoIP, o si se encuentran cómodos con el uso de gateways (enlaces de conmutación) de medios externos para conectar aparatos telefónicos convencionales, entonces usted tenemos la opción de emplearlos también en distribuciones libres como FreeBSD, Mac OS X, NetBSD, OpenBSD y Solaris; inclusive es compatible con Windows como podemos observar la consola de Asterisk en Windows en la figura 2.4. Para ello debes configurar las opciones desde el puerto para controladores de Digium y seleccionar las plataformas diferentes a Linux.



```
AsteriskWin32 PBX
Main Status PBX Tools Help

[ Reading Master Configuration ][ Initializing Custom Configuration Options]warning! Asterisk is not t
Asterisk 1.0.10, Copyright (C) 1999-2004 Digium.
Written by Mark Spencer <markster@digium.com>
=====
[ Booting...Aug 4 13:57:07 WARNING[2868]: pbx.c:5093 ast_context_verify_includes: Context 'de
Aug 4 13:57:07 WARNING[2868]: pbx.c:5093 ast_context_verify_includes: Context 'local' tries inc
.....Aug 4 13:57:08 ERROR[2868]: chan_t
Aug 4 13:57:08 ERROR[2868]: chan_tapi.c:832 load_module: Unable to register device 'auto'
Aug 4 13:57:08 WARNING[2868]: chan_capi.c:3237 load_module: CAPI not installed. CAPI disab
]
Asterisk Ready.

Asterisk Ready
```

Fig. 2.4 Interfaz de Asterisk Win32PBX

La compatibilidad mayor se obtiene cuando se emplea en combinación con Linux Kernel 2.4 para una arquitectura Debian i386 siendo compatible con Fedora, Gentoo, Mandrake: 9.0, CentOS, Pie Box, Tao Linux, Whitebox, Slackware 9.0, 9.1, 10.0 y 10.2, SuSE: Version 8, 9, 10.0 y 10.1, Trustix version 2.2, Yellow Dog para plataformas PPC, YDL 3.01, UltraLinux Sparc64 y Gentoo/Sparc64, Red Hat.

Este software es compatible con los siguientes sistemas de telefonía: telefonía ATCOM IP AT-510/AT530, teléfonos Aastra – Sayson, adaptadores analógicos para teléfonos Cisco 79xx series, Cisco ATA 18x y Cisco 12SP+/VIP30, Cortelco 2747 tricks, GNET VP320 basado en el chip PA1688, Grandstream BudgeTone, Grandstream GXP2020, Linksys SPA-941, teléfonos Mitel series 5055, 5215, 5220, teléfonos Nortel, ShoreTel 210, Siemens HiNet LP5100, Siemens OptiPoint 600, Siemens Gigaset DECT, Sipura SPA-2000 y SPA-3000, Swissvoice IP10s, Soyo G668, Uniden UIP200, Pulverinnovations WISIP, tel130fonos Zultys, Zyxel P2000W y VTA1000.

2.2 FXS Y FXO

FXS y FXO son los nombres de los puertos usados por las líneas telefónicas analógicas (también denominados POTS - Servicio Telefónico Básico y Antiguo)

2.1.3 FXS

La interfaz de abonado externo es el puerto que efectivamente envía la línea analógica al abonado. En otras palabras, es el “enchufe de la pared” que envía tono de marcado, corriente para la batería y tensión de llamada

2.1.4 FXO

Interfaz de central externa es el puerto que recibe la línea analógica. Es un enchufe del teléfono o aparato de fax, o el enchufe de su centralita telefónica analógica. Envía una indicación de colgado/descolgado (cierre de bucle). Como el puerto FXO está adjunto a un dispositivo, tal como un fax o teléfono, el dispositivo a menudo se denomina “dispositivo FXO”.

Son siempre pares, es decir, similar a un enchufe macho/hembra.

Sin una centralita, el teléfono se conecta directamente al puerto FXS que brinda la empresa telefónica un ejemplo se muestra en la figura 2.5.

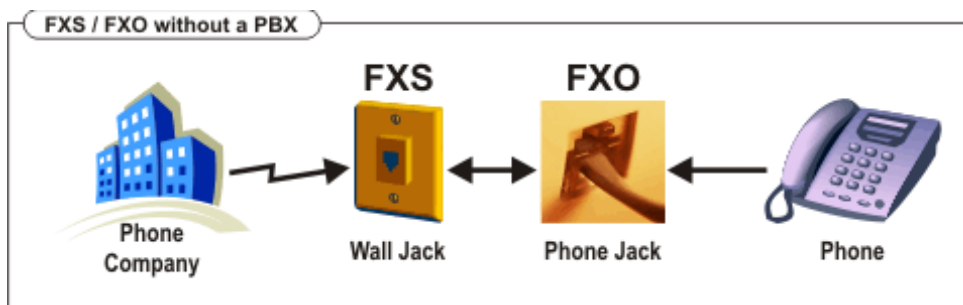


Fig. 2.5 FXS/FXO SIN PBX

Si tiene centralita, como se muestra en la figura 2.6 se debe conectar las líneas que suministra la empresa telefónica a la centralita y luego los teléfonos a la centralita. Por lo tanto, la centralita debe tener puertos FXO (para conectarse a los puertos FXS que suministra la empresa telefónica) y puertos FXS (para conectar los dispositivos de teléfono o fax).

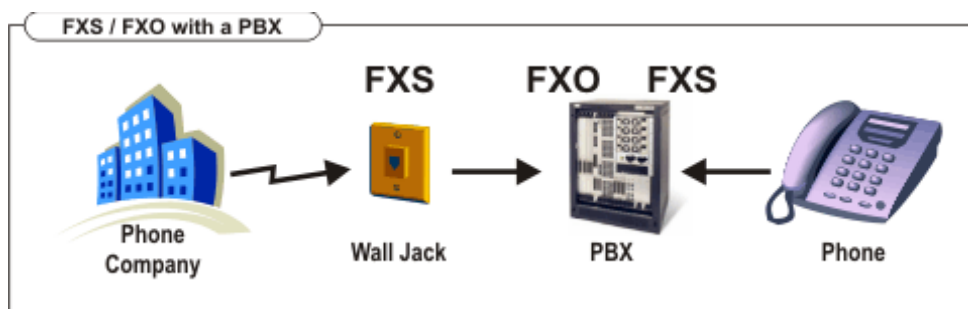


Figura2.6 FXS/FXO CON PBX

2.3 CHAN MOBILE

2.3.1 Historia de Chan Mobile

En de las primeras versiones se utilizaba el paquete `chan_bluetooth` de Theo Zourzouvillys la cual estaba en cierto sentido desactualizada, durante algunos meses atrás en el trunk de Asterisk se comenzó con el desarrollo del software `chan_cellphone` el cual derivo en la versión que actualmente se conoce como `chan_mobile` y que está disponible vía SVN y en la versión 1.6.x de Asterisk.

2.3.2 Chan Mobile Trabajando con Asterisk

La meta de nuestro proyecto es enrutar las llamadas desde los anexos internos IP de un servidor con Asterisk hacia la red GSM y vice versa usando un canal Bluetooth además de la posibilidad de enviar y recibir mensajes de texto SMS.

La idea del sistema es poder a través del dial plan generar una llamada hacia un dongle bluetooth (o adaptador USB Bluetooth) el cual debe estar pareado con algún equipo teléfono celular, por lo tanto la llamada será gestionada por el canal bluetooth y el celular tendrá que soportar el modo Audio Gateway para tal efecto. Aunque se ha probado en varios modelos de equipo celulares este desarrollo en algunos casos no está soportado debido a que el equipo no posee esa propiedad.

Chan mobile fue escrito por David Bowerman y está soportado oficialmente sólo por el tronco de desarrollo de Asterisk y está disponible como un add-on de <http://svn.digium.com/svn/>.

La documentación actual se puede obtener de <http://svn.digium.com/svn/>.

El controlador canal asterisk permite usar teléfonos móviles como dispositivos FXO y auriculares como dispositivos FXS.

2.3.3 Características

- Múltiples teléfonos celulares pueden ser conectados.
- Múltiples adaptadores bluetooth pueden ser conectados.
- Asterisk se conecta automáticamente a cada teléfono celular cuando viene en un rango.
- Tiene comando para detectar dispositivos Bluetooth. Útiles para la configuración. Necesita un adaptador Bluetooth desocupado.
- Las llamadas entrantes a los teléfonos celulares son manejadas por Asterisk, al igual que las llamadas entrantes en un canal Zap.
- Identificación de llamadas a través de las llamadas entrantes.
- Dial de salida en un teléfono celular con acceso telefónico (celular / dispositivo / nnnnnnn) en el dialplan.
- Puede usar un auricular Bluetooth como la extensión con Acceso telefónico (celular / dispositivo) en el dialplan.
- La aplicación CellStatus puede ser utilizada en el dialplan para ver si está conectado un teléfono celular.
- Existe la aplicación MobileSMS para enviar SMS a través de un teléfono móvil conectado.

Usar este modulo también implica tener precauciones y cuidados, por ejemplo:

- No todos los móviles con los perfiles de Bluetooth tienen todas las

características necesarias para este canal de trabajo. Por ejemplo - no se puede utilizar Nokia E65.

- Se necesita usar un dongle bluetooth por celular al momento de la conexión con el servidor Asterisk.

La mayoría de la información disponible ya es de configuración pero va en otra sección de este documento instalación, configuración e implementación, etc.

CAPITULO 3

IMPLEMENTACION

3.1 INTRODUCCION

El fin del proyecto de graduación es conocer y aplicar las bondades del software libre, ya que nos permite disminuir costos al implementar soluciones en la vida real.

Esta solución está basada en el software libre asterisk que permite implementar centrales telefónicas a pequeña, mediana y gran escala, que por ser de libre acceso lo utilizamos en forma gratuita. Además tiene muchos addons que nos sirven para ampliar los usos de dicho software y encontrar recursos para solucionar toda clase de problemas como lo resolverían las centrales telefónicas privadas, con la diferencia que ahorramos el costo del equipo.

La central telefónica tradicional es reemplazada por un computador que variará en su costo dependiendo de cuán grande necesitemos sea nuestra central y el tamaño la central dependerá de la concurrencia de llamadas que vaya a tener y muchos otros factores, pero siempre el dinero gastado al final será menor que si compráramos una central propietaria.

3.2 HARDWARE

Las características para seleccionar un servidor se puede considerar una tarea sencilla porque puede ser cualquier equipo basado en plataforma x86, y

a la vez complicada, porque el rendimiento de su sistema dependerá de la atención que le ponga a la plataforma de implementación. Cuando seleccione el hardware, usted debe considerar cuidadosamente el diseño general de su sistema y las funcionalidades que requerirá, esto le ayudará a determinar la marca y modelo del CPU, tarjeta madre, y fuente de energía.

3.2.1 Servidor

Los requisitos de hardware que se han configurado en cada uno de los servidores para el desarrollo del proyecto, son los siguientes:

Tabla I Características del Servidor

Procesador	Intel Pentium 4 de 3GHZ
RAM	1.5 GB
Disco Duro	20GB
Tarjeta de Red	10/100 Mbps

3.2.2 Bluetooth

La conexión del Servidor Asterisk con los Gateways se dará por medio de un bluetooth dongle de las siguientes características

Tabla II Bluetooth Dongle

Bluetooth Dongle	Omega EDR Class II
Distancia	100 metros sin obstáculo
Tasa Transferencia	723KBPS/ACL 64Kbps/SCO
Frecuencia	2.400~2.4835GHZ Banda ISM

3.2.3 Gateways

Para realizar las conexiones con la red GSM también es necesario un dispositivo móvil con tecnología bluetooth, en este proyecto se han implementado 3 gateways cuyos modelos se detallan a continuación:

Tabla III Gateways

Gateway	Marca	Modelo
Gateway 1 : Porta	Motorola	V3
Gateway 2 : Movistar	Nokia	6230
Gateway 3 : Alegro PCS	Nokia	3120 classic

3.2.4 Teléfonos IP

- **GXP1200**



Fig. 3.1 Teléfono GXP1200

Este es el teléfono que hemos utilizado para las pruebas en casa, ya que este teléfono brinda las mismas bondades que el teléfono ocupado en los laboratorios.

- **GXP2000**



Fig. 3.2 Teléfono GXP2000

Teléfono de utilizado en las clases prácticas del laboratorio de telecomunicaciones.

3.3 SOFTWARE

3.3.1 Servidor PBX

El servidor que será utilizado como centralita telefónica tendrá los siguientes componentes instalados:

Tabla IV Servidor PBX

Sistema Operativo	Red Hat Enterprise Linux 5
Software IP PBX	Asterisk versión 1.6.0.10
Add-on Chan_mobile	Asterisk Addons 1.6.0.3

Librerías necesarias para que Asterisk funcione correctamente como centralita PBX

kernel	zlib-devel
kernel-devel	openssl
bison	openssl-devel
bison-devel	gnutls-devel
ncurses	gcc
ncurses-devel	gcc-c++
zlib	

3.3.2 BlueZ

BlueZ proporciona soporte para el núcleo Bluetooth y los protocolos de capas. Es flexible, eficiente y utiliza una aplicación modular.

Librerías necesarias:

- bluez-utils
- bluez-libs
- bluez-libs-devel

3.3.3 Softphones

Los softphones son simuladores de extensiones de una centralita telefónica, en este proyecto se han utilizado dos tipos, uno para las extensiones que usara el protocolo SIP y otro que simulara extensiones IAX.

Simulador extensiones SIP:

- X-Lite

Simulador extensiones IAX:

- Zoiper

3.4 INSTALACIÓN

3.4.1 Instalación de Librerías Base

Ejecutar los siguientes comandos.

```
yum install kernel-devel
```

```
yum install kernel
```

```
yum install bison
```

```
yum install bison-devel
```

```
yum install ncurses
```

```
yum install ncurses-devel
```

```
yum install zlib
```

```
yum install zlib-devel
```

```
yum install openssl
```

```
yum install openssl-devel
```

```
yum install gnutls-devel
```

```
yum install gcc
```

```
yum install gcc-c++
```

3.4.2 Instalación de Librerías para Bluetooth

Ejecutar los siguientes comandos.

```
yum install bluez-utils
```

```
yum install bluez-libs
```

```
yum install bluez-libs-devel
```

3.4.3 Instalación de Asterisk

Descargar los paquetes del software IP PBX, Asterisk y del Asterisk Addons 1.6.0.3 a la carpeta /usr/src. Luego de descargarlos ejecutar los siguientes comandos:

Nos ubicamos en el directorio /usr/src ejecutando:

```
cd /usr/src
```

Descomprimos los archivos descargados previamente con los siguientes comandos

```
tar -xvzf asterisk-1.6.0.10.tar.gz
```

```
tar -xvzf asterisk-addons-1.6.0.3.tar.gz
```

Accedemos al directorio donde se descomprimió asterisk-1.6.0.10

```
cd asterisk-1.6.0.10
```

Instalamos asterisk ejecutando los siguientes comandos secuencialmente

```
./configure
```

```
make
```

```
make install
```

```
make config
```

Si deseamos crear configuraciones de ejemplo podemos ejecutar el siguiente comando opcional

```
make samples
```

Regresamos al directorio donde descargamos los paquetes instaladores

```
cd /usr/src
```

Ingresamos a la carpeta donde se descomprimieron los instaladores de los addons de asterisk:

```
cd asterisk-addons-1.6.0.10
```

Para instalar los addons de asterisk en el cual está incluido el modulo chan_mobile escribimos lo siguiente en la línea de comando:

```
./configure -with-bluetooth
```

```
make
```

```
make install
```

```
make samples
```

La instalación creara una carpeta llamada asterisk en el directorio /etc de esta manera los archivos de configuración de asterisk se encuentran bajo este directorio:

/etc/asterisk/

3.5 CONFIGURACION DE ARCHIVOS DE ASTERISK

3.5.1 Configuración SIP.CONF

/etc/asterisk/sip.conf

En este archivo se configuran todas la extensiones que van a usar el protocolo SIP.

El archivo sip.conf está estructurado en tres bloques. **La parte general** donde hay que definir la configuración general de nuestras extensiones, el bloque central donde configuraremos la registración a nuestros proveedores VoIP (y, si queremos, los datos para conectar entre ellos distintos servidores Asterisk), y **la parte final** del archivo donde configuraremos todas nuestras extensiones internas y externas.

3.5.1.1 Configuración general

[general]

Etiqueta que introduce la parte general de la configuración.

context=default

Permite hacer búsquedas de registros DNS SRV para llamadas SIP salientes basadas en los nombres de dominio

svrlookup=yes

Permite hacer búsquedas de registros DNS SRV para llamadas SIP salientes basadas en los nombres de dominio.

language=es

Si hemos instalado locuciones en más de un idioma, aquí podemos definir cual idioma usará la extensión. En este caso se instalaron las locuciones en español, y se lo especifica con el prefijo es.

disallow=all
allow=gsm
allow=ulaw

Mediante las líneas anteriores se configuran los códecs a usar. Se usarán el GSM y el ULAW para compatibilidad con softphones.

Configuracion final:

[general]
context=default
svrlookup=yes
language=es
disallow=all
allow=gsm
allow=ulaw

3.5.1.2 Configuración extensiones

[1001]

Número de la extensión

type=friend

Tipo de extensión. Puede ser friend, user o peer. Friend puede hacer y recibir llamadas, user solo recibir y peer solo puede hacer (como en el caso de proveedores VoIP que usamos solo para hacer llamadas)

secret=1001

Define la contraseña de la extensión

qualify=yes

Determina el tiempo de respuesta de una extensión y si está alcanzable o no

nat=no

Si la extensión se conecta al servidor asterisk detrás de un firewall hay que poner yes, caso contrario no.

host=dynamic

si la extensión se conecta remotamente cambiando continuamente su dirección IP se pone este parámetro

canreinvite=no

Yes si queremos que la extensión intente conectarse directamente con la extensión llamada. No si queremos que Asterisk haga de puente entre las dos extensiones.

context=internal

El contexto que usará la extensión

Configuración final de 4 extensiones SIP:

***[1001]
type=friend
secret=1001
qualify=yes
nat=no
host=dynamic
canreinvite=no
context=internal***

***[1002]
type=friend
secret=1002
qualify=yes
nat=no
host=dynamic
canreinvite=no
context=internal***

***[1003]
type=friend
secret=1003
qualify=yes
nat=no
host=dynamic
canreinvite=no
context=internal***

***[1903]
type=friend
secret=1903
qualify=yes
nat=no
host=dynamic
canreinvite=no
context=internal***

3.5.2 Configuración EXTENSIONS.CONF

/etc/asterisk/extension.conf

Este archivo contiene el plan de marcado de la central telefónica.

El archivo extensions.conf es el más importante del Asterisk y tiene como misión principal definir el dialplan o plan de numeración que seguirá la centralita para cada contexto y por tanto para cada usuario.

El fichero extensions.conf se compone de secciones o contextos entre corchetes []. Hay dos contextos especiales que están siempre presentes que son [general] y [globals]

3.5.2.1 Contexto General

[general]

Se establecen configuraciones generales. Que se aplica al resto de contexto.

static=yes

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

Configuración final contexto general:

[general]
static=yes

3.5.2.2 Contexto global

En este contexto se establecen las variables globales para el dial plan.

[globals]

CELLG1_MIN=1

Define el mínimo número de grupo para Porta (grupo 1).

CELLG1_MAX=10

Define el máximo número de grupo para Porta (grupo 1).

CELLG2_MIN=11

Define el mínimo número de grupo para Movistar (grupo 2)

CELLG2_MAX=20

Define el máximo número de grupo para Movistar (grupo 2),

CELLG3_MIN=21

Define el mínimo número de grupo para Alegro (grupo 3).

CELLG3_MAX=30

Define el máximo número de grupo para Alegro (grupo 3).

CELLSMS=movistar

Indica el nombre del dispositivo móvil configurado en */etc/asterisk/mobile.conf* que servirá como Gateway para el envío de mensajes. En este caso será el nombrado *movistar*.

CELLG1_MAX_CALLS=1

Define el Número máximo de llamadas simultáneas en grupo 1.

CELLG2_MAX_CALLS=1

Número máximo de llamadas simultáneas en grupo 2.

CELLG3_MAX_CALLS=1

Define el número máximo de llamadas simultáneas en grupo 3.

MAX_TIME=70000

Define el máximo tiempo de duración de una llamada por celular en milisegundos (el tiempo cuenta desde que se empieza a llamar para *chan_mobile*).

WAIT_TIME=15

Define el tiempo que debe esperar una persona en segundos antes de volver a intentar llamar por una línea ocupada.

MAX_RETRIES=3

Define el Número máximo de intentos antes de colgar a una persona que está intentando acceder a una línea celular.

PEXT_MIN=1900

Privileged extensions: Limite inferior de extensiones privilegiadas.

PEXT_MAX=1999

Privileged extensions: Limite superior extensiones privilegiadas.

LIMIT_WARNING_FILE=beep

Parámetro de la función de Asterisk Dial. Indica un sonido para cuando esté próximo a terminar la llamada.

3.5.2.3 Contexto internal

En este contexto se establece el dial plan para las extensiones internas que se conectan mediante el protocolo SIP.

exten => _1XXX,1,Dial(SIP/\${EXTEN},15)

Define el canal de salida SIP para las extensiones desde la 1000 hasta la 1999

```
exten => _1XXX,n,Playback(cm-nodisponible)
```

De no contestar la extensión destinataria el usuario escuchara un mensaje indicando que la extensión a la cual está llamando no se encuentra disponible.

```
exten => _1XXX,n,Hangup()
```

Se colgara, así terminado la llamada.

```
include => outgsm
```

Se incluye el contexto outgsm.

3.5.2.4 Contexto outgsm

Define el dial plan para las llamadas salientes hacia la red gsm

```
[outgsm]
```

```
Define el contexto
```

```
include => porta
```

Incluye el contexto porta

```
include => movistar
```

Incluye el contexto movistar

```
include => alegre
```

Incluye el contexto alegre

include => mensajes

Incluye el contexto para el manejo de mensajes

3.5.2.5 Contexto mensajes

Para el envío de mensajes, se estableció un código el cual consta de la extensión para mensajes, el tipo de mensaje, y los parámetros de envío como hora, fecha y número de celular según cada mensaje.

Para coger cada uno de los datos, se lo realiza con la variable `#{EXTEN}` en la cual se almacena la extensión marcada por el usuario. También se hace uso de la función *MobileStatus*, esta nos permitirá saber el estado del celular; *MobileStatus* pertenece al modulo *chan_mobile*, recibe como parámetros el nombre del celular del cual queremos conocer su estado, y que para nuestro caso se lo ha declarado globalmente y almacenado en la variable *CELLSMS* y como segundo parámetro el nombre de la variable en la cual deseamos guardar el estado del celular a la que hemos llamado *SMSTATUS*.

El envío de mensajes se realiza de acuerdo al siguiente diagrama de flujo:



Fig. 3.3 Diagrama de flujo para envío de SMS

Con este diagrama podemos observar que al marcar la extensión para mensajería, lo primero que se verifica es la disponibilidad del celular gateway, dependiendo de eso se realiza el envío cuando este se encuentre disponible, y se colgará la llamada en caso contrario.

La plantilla de mensajes se muestra en la tabla 5:

Tabla V Plantilla de Mensaje

<i>Mensaje 1</i>	<i>La Reunión está establecida para el DD/MM a las HH:MM</i>
<i>Mensaje 2</i>	<i>Reunión se ha cancelado</i>
<i>Mensaje 3</i>	<i>No Me encuentro disponible, Llámeme el DD/MM a las HH:MM</i>
<i>Mensaje 4</i>	<i>Favor, llame al XXXXXXXXXX</i>
<i>Mensaje 5</i>	<i>Llego tarde, Estare ahi a las HH:MM</i>

El dial plan se maneja por prioridades, siendo 1 la prioridad más alta y 2, 3, 4,5... n etc. las prioridades siguientes. Si hay disponibilidad, dependiendo del valor de la variable SMSTATUS se salta de la prioridad 1 a la prioridad 3 la cual realiza el envío del mensaje de texto, sin embargo si el móvil no se encuentra disponible se salta a la prioridad 7 la que indica al usuario que el móvil no se encuentra disponible.

Para la configuración del dial plan de los mensajes de texto sigue el siguiente procedimiento:

1. Determinar el Estado del móvil Gateway de mensajería.
2. Si está disponible sigue con la prioridad 3 caso contrario salta a la prioridad 7.
3. Se define la prioridad 3, envío del mensaje de texto.
4. Se define prioridad 4 la cual da a conocer el usuario que se envió el mensaje de texto.
5. Se define la prioridad 7 que da a conocer al usuario que el móvil no se encuentra disponible.

Tabla de parámetros para plantilla de mensaje 1:

Tabla VI Plantilla de Mensaje 1

<i>Mensaje 1</i>	
Mensaje	<i>La Reunión esta establecida para el 21/08 a las 21:30</i>
Extensión	0
Tipo	01
Destinatario	XXXXXXXXXX
Fecha DDMM	XXXX
Hora HHMM	XXXX
Celular	-

Configuración para plantilla de mensaje 1:

[mensajes]

**exten => _001XXXXXXXXXXXXXXXXXXXX,1,
MobileStatus(\${CELLSMS},SMSTATUS)**

exten => _001XXXXXXXXXXXXXXXXXXXX,2, Gotof(\$[\${SMSTATUS}=2]?3:7)

**exten => _001XXXXXXXXXXXXXXXXXXXX,3,
MobileSendSMS(\${CELLSMS},\${EXTEN:3:9},La Reunion esta establecida para el
\${EXTEN:12:2}/\${EXTEN:14:2} a las \${EXTEN:16:2}:\${EXTEN:18:2})**

exten => _001XXXXXXXXXXXXXXXXXXXX,4,Playback(cm-smsexitoso)

exten => _001XXXXXXXXXXXXXXXXXXXX,7,Playback(cm-smsfallido)

Tabla de parámetros para plantilla de mensaje 2:

Tabla VII Plantilla de Mensaje 2

<i>Mensaje 2</i>	
Mensaje	<i>Reunión se ha cancelado</i>
Extensión	0
Tipo	02
Fecha DDMM	XXXXXXXXXX
Hora HHMM	-
Hora	-
Celular	-

Configuración para plantilla de mensaje 2:

```
exten => _002XXXXXXXXXX,1,  
MobileStatus(${CELLSMS},SMSTATUS)  
  
exten => _002XXXXXXXXXX,2, Gotolf(${SMSTATUS}=2]?3:7)  
  
exten => _002XXXXXXXXXX,3,  
MobileSendSMS(${CELLSMS},${EXTEN:3:9},La Reunion se ha cancelado)  
  
exten => _002XXXXXXXXXX,4,Playback(cm-smsexitoso)  
exten => _002XXXXXXXXXX,7,Playback(cm-smsfallido)
```

Tabla de parámetros para plantilla de mensaje 3:

Tabla VIII Plantilla de Mensaje 3

<i>Mensaje 3</i>	
Mensaje	<i>No Me encuentro disponible, Llámeme el 21/08 a las 21:30</i>
Extensión	0
Tipo	03
Fecha DDMM	XXXXXXXXXX
Hora HHMM	XXXX
Hora	XXXX
Celular	-

Configuración para plantilla de mensaje 3:

```

exten => _003XXXXXXXXXXXXXXXXXXXX,1, MobileStatus(${CELLSMS},SMSTATUS)
exten => _003XXXXXXXXXXXXXXXXXXXX, 2, Gotolf(${SMSTATUS}=2]?3:7)
exten => _003XXXXXXXXXXXXXXXXXXXX,3, MobileSendSMS(${CELLSMS},
${EXTEN:3:9}, No Me encuentro disponible, Llámeme el
${EXTEN:12:2}/${EXTEN:14:2} a las ${EXTEN:16:2}:${EXTEN:18:2} )

exten => _003XXXXXXXXXXXXXXXXXXXX,4,
Playback(sms-exitoso)

exten => _003XXXXXXXXXXXXXXXXXXXX,7,
Playback(cm-smsfallido)

```

Tabla de parámetros para plantilla de mensaje 4:

Tabla IX Plantilla de Mensaje 4

<i>Mensaje 4</i>	
Mensaje	<i>Favor, llame al 094834663</i>
Extensión	0
Tipo	04
Fecha DDMM	XXXXXXXXXX
Hora HHMM	-
Hora	-
Celular	XXXXXXXXXX

Configuración para plantilla de mensaje 4:

```
exten => _004XXXXXXXXXXXXXXXXXXXX,1,  
MobileStatus(${CELLSMS},SMSTATUS)  
  
exten => _004XXXXXXXXXXXXXXXXXXXX,2, Gotof(${SMSTATUS}=2)?3:7)  
  
exten => _004XXXXXXXXXXXXXXXXXXXX,3,  
MobileSendSMS(${CELLSMS},${EXTEN:3:9},Por Favor, llame al ${EXTEN:12:9})  
  
exten => _004XXXXXXXXXXXXXXXXXXXX,4,  
Playback(cm-smsexitoso)  
  
exten => _004XXXXXXXXXXXXXXXXXXXX,7,  
Playback(cm-smsfallido)
```

Tabla de parámetros para plantilla de mensaje 5:

Tabla X Plantilla de Mensaje 5

<i>Mensaje 5</i>	
Mensaje	<i>Llego tarde, Estare ahi a las 21:30</i>
Extensión	0
Tipo	05
Destinatario	XXXXXXXXXX
Fecha DDMM	-
Hora HHMM	XXXX
Celular	-

Configuración para plantilla de mensaje 5:

```

exten => _005XXXXXXXXXXXXXXXX,1,
MobileStatus(${CELLSMS},SMSTATUS)

exten => _005XXXXXXXXXXXXXXXX,2, GotIIf($[${SMSTATUS}=2]?3:7)

exten => _005XXXXXXXXXXXXXXXX,3,
MobileSendSMS(${CELLSMS},${EXTEN:3:9},Llego tarde, Estare ahi a las
${EXTEN:12:2}:${EXTEN:14:2})

exten => _005XXXXXXXXXXXXXXXX,4,
Playback(cm-smsexitoso)

exten => _005XXXXXXXXXXXXXXXX,7,Playback(cm-smsfallido)

```

3.5.2.6 Contextos de Operadores Celulares

Para una mejor administración, se ha separado los números de celulares las operadoras en contextos diferentes para cada una. La forma de conocer la red a la que pertenecen los distintos números telefónicos de celulares es por medio del plan numérico establecido por el estado ecuatoriano a través de la Superintendencia de Telecomunicaciones.

En la siguiente tabla se muestran el plan de numeración que rige en el Ecuador:

Tabla XI Series Numéricas de celulares Alegro

SERIES NUMERICAS MOVIL			
OPERADOR	SERIE ASIGNADA		
	PREFIJO	1ºDIGITO	2º DIGITO
ALEGRO TELECSA	8	2	0
			1
			2
			3
			4
	9	9	0
			1
			2
			3
			4
			5
			6
			7
			8
			9

Tabla XII Series Numéricas de Celulares Movistar

SERIES NUMERICAS			
OPERADOR	SERIE ASIGNADA		
	PREFIJO	1º DIGITO	2ºDIGITO
MOVISTAR OTECEL	8	3, 4, 7	0
			1
			2
			3
			4
			5
			6
			7
			8
			9
	9	5, 8	0
			1
			2
			3
			4
			5
			6
			7
			8
			9
	9	2	5
			6
			7
			8
			9
	9	9	0
			2
			7
			8
			9

Tabla XIII Series Numéricas de Celulares Porta

SERIES NUMERICAS			
OPERADOR	SERIE ASIGNADA		
	PREFIJO	1ºDIGITO	2ºDIGITO
PORTA CONECEL	8	2	5
			6
			7
			8
			9
	8	0, 1, 5, 6, 8, 9	0
			1
			2
			3
			4
			5
			6
			7
			8
			9
	9	0, 1, 3, 4, 7	0
			1
			2
			3
			4
			5
			6
			7
			8
			9
	9	2	0
			1
			2
			3
			4
	9	9	1
			3
			4
			5
			6

En la implementación de los contextos por operador, se realizan usando las tablas anteriormente expuestas y para número celular se ejecuta la macro `outgsm_macro`, la cual será explicada en detalle en el numeral siguiente.

A continuación la implementación de los contextos:

```
[porta]
exten => _908[015689]XXXXXX,1,Macro(outgsm_macro, CELLG1)
exten => _9082[5-9]XXXXXX,1,Macro(outgsm_macro, CELLG1)
exten => _909[01347]XXXXXX,1,Macro(outgsm_macro, CELLG1)
exten => _9092[0-4]XXXXXX,Macro(outgsm_macro, CELLG1)
exten => _9099[13456]XXXXXX,Macro(outgsm_macro, CELLG1)

[movistar]
exten => _908[347]XXXXXX,1,Macro(outgsm_macro, CELLG2)
exten => _9092[5-9]XXXXXX,1,Macro(outgsm_macro, CELLG2)
exten => _909[58]XXXXXX,1,Macro(outgsm_macro, CELLG2)
exten => _9099[02789]XXXXXX,1,Macro(outgsm_macro, CELLG2)

[alegro]
exten => _9082[0-4]XXXXXX,1,Macro(outgsm_macro, CELLG3)
exten => _9096XXXXXX,1,Macro(outgsm_macro, CELLG3)
```

3.5.2.7 Macro OUTGSM_MACRO

Esta macro es la que realiza el procedimiento para conectarse a la red GSM, maneja colas de espera y regula la duración de las llamadas. El diagrama de flujo de la figura 3.2 muestra el procedimiento que sigue esta macro.

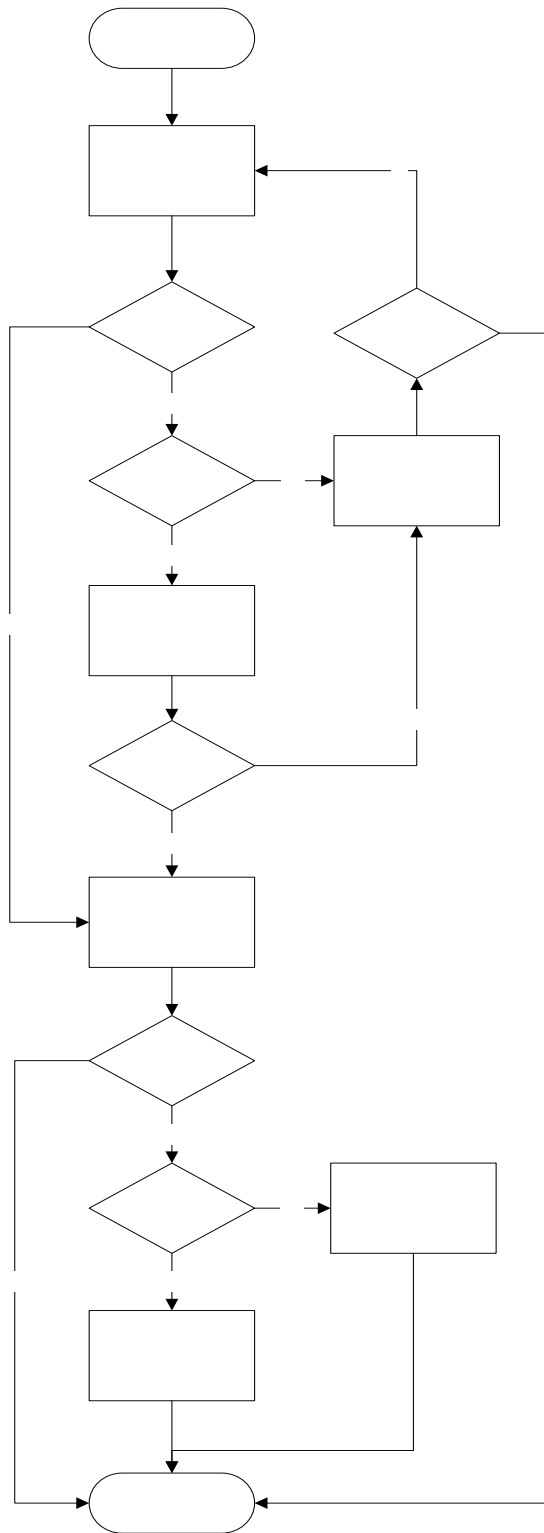


Fig. 3.4 Diagrama de macro para conectarse a RED GSM

Inicialmente cuando llega una llamada saliente, se determina el canal de salida según la operadora celular del número de destino. Luego se averigua si hay una cola de espera por ese canal, es decir ya se está usando y puede o no haber otras llamadas esperando por ese mismo canal. Si no es así, la llamada accede directamente al dispositivo usado como Gateway para realizar la llamada, caso contrario se pregunta si la llamada provino de una de las extensiones privilegiadas. Si lo anterior es verdadero se pregunta por todos los canales de cualquier operadora hasta encontrar alguno disponible que pueda realizar la llamada y, si todos se encuentran ocupados, se coloca a la llamada saliente en un estado de espera. Las llamadas que no pertenezcan al rango de extensiones privilegiadas irán directamente al estado de espera.

En el estado de espera las llamadas deberán aguardar un tiempo definido en la variable global *WAIT_TIME*, el mismo que empieza al instante del ingreso de la llamada al estado de espera. Al final de ese periodo se consultará si ya se ha alcanzado el número máximo de intentos permitidos por llamada, el cual se define en la variable global *MAX_RETRIES*; si es cierto, se colgará la llamada, sino lo es, se incrementará el contador de intentos y se intentará nuevamente acceder al canal correspondiente, volviéndose a realizar el proceso del párrafo anterior.

Al momento de realizar la llamada, es decir cuando el canal respectivo está disponible, se establece el tiempo máximo de la duración de la llamada si esta proviene de una extensión normal o, si es privilegiada, se asigna tiempo ilimitado. Si el destinatario no está disponible, se cuelga la llamada.

Esta macro utiliza un solo argumento:

ARG1

El grupo en el cual se alojará la llamada saliente según la operadora de destino.

Para el funcionamiento de `outgsm_macro` se hace uso también de algunas variables de canal, las cuales permiten parametrizar el comportamiento específico del manejo de llamadas.

PEXT

Valor booleano que indica si la llamada provino de una extensión privilegiada.

INTENTOS

Contador que lleva registro del número de veces que se ha intentado acceder al canal de comunicación con la red GSM.

GRUPO

Define el grupo de celulares por donde se realizará la llamada. Es de principal uso cuando la llamada proviene de una extensión privilegiada y la llamada se realizaría en un grupo distinto al que normalmente le tocaría

DIALOPT

Define las opciones que se establecerán en la función Dial, particularmente en lo referente a la duración de las llamadas.

i

Variable para iteración en la selección de gateway de salida según la disponibilidad, en un mismo grupo de celulares.

En las siguientes líneas se detalla la implementación completa de la macro outgsm_macro.

```
exten => s,1,Set(PEXT=0)
exten => s,n,Gotolf(${CALLERID(num)} >= ${PEXT_MIN} && ${CALLERID(num)} <=
${PEXT_MAX}]?:pext_ig0)
exten => s,n,Set(PEXT=1)
exten => s,n(pext_ig0),Set(INTENTOS=1)

exten => s,n(askchannel),Set(GRUPO=${ARG1})
exten => s,n,Gotolf(${GROUP_COUNT(${GRUPO})} >= ${${GRUPO}_MAX_CALLS}]?:acc_canal)
exten => s,n,Gotolf(${PEXT}?:esperar)

exten => s,n,Gotolf(${GROUP_COUNT(CELLG1)} >= ${CELLG1_MAX_CALLS}]?:sig_g2:)
exten => s,n,Set(GRUPO=CELLG1)
exten => s,n,Goto(acc_canal)

exten => s,n(sig_g2),Gotolf(${GROUP_COUNT(CELLG2)} >= ${CELLG2_MAX_CALLS}]?:sig_g3:)
```

```

exten => s,n,Set(GRUPO=CELLG2)
exten => s,n,Goto(acc_canal)

exten => s,n(sig_g3),GotoIf(${GROUP_COUNT(CELLG3)} >= ${CELLG3_MAX_CALLS})?esperar:)
exten => s,n,Set(GRUPO=CELLG3)
exten => s,n,Goto(acc_canal)

exten => s,n(acc_canal),Set(GROUP(=${GRUPO})
exten => s,n,Set(i=${GRUPO}_MIN)
exten => s,n,Set(DIALOPT=rL(${MAX_TIME}:10000))
exten => s,n,GotoIf(${PEXT}?dialcell)
exten => s,n,Set(DIALOPT=r)
exten => s,n(dialcell),Dial(Mobile/g${i}/${MACRO_EXTEN:1},, ${DIALOPT})
exten => s,n,Set(i=${i}+1)
exten => s,n,GotoIf(${i} > ${GRUPO}_MAX)?nodisponible:)
exten => s,n,Goto(dialcell)
exten => s,n(nodisponible),NoOp(${DIALSTATUS})
exten => s,n,Playback(cm-nodisponible)
exten => s,n,Hangup()

exten => s,n(esperar),GotoIf(${INTENTOS} > ${MAX_RETRIES})?fin_intentos:)
exten => s,n,Set(INTENTOS=${1 + ${INTENTOS}})
exten => s,n,Playback(cm-ocupado)
exten => s,n,WaitMusicOnHold(${WAIT_TIME})
exten => s,n,Goto(askchannel)
exten => s,n(fin_intentos),Playback(cm-finintentos)
exten => s,n,Hangup()

```

3.5.3 Configuración MOBILE.CONF

Mobile.conf es el archivo que contiene las configuraciones de los parámetros necesarios que se requieren para poder conectar un dispositivo móvil mediante bluetooth con el servidor asterisk.

En este archivo se necesita información como: Mac Address del dispositivo dongle que permite al servidor asterisk conectarse a los gateways así como también la dirección Mac de cada celular que va a establecer un canal de conexión, a estos también se asocia un

puerto y definir un adaptador por el cual se establecerán dichas conexiones.

3.5.3.1 Contexto adapter

Aquí se listan los adaptadores que se usarán, deben poseer un id único, y la dirección es la dirección mac del dispositivo. Cabe destacar que el uso de los adapters es de uno por dispositivo conectado.

```
[adapter]  
id=blue  
address=00:11:67:D6:38:2E
```

Se define el id blue y la MAC address del adaptador bluetooth.

3.5.3.2 Contexto de dispositivo

El nombre de este contexto se puede elegir según la conveniencia.

```
[porta]  
address=00:17:84:4E:91:FA  
port=7  
context=incoming-mobile  
adapter=blue  
nocallsetup=yes  
group=1
```

Esta configuración hace relación a los parámetros de configuración necesarios para los dispositivos móviles. El puerto de cada celular se lo puede descubrir mediante el comando **mobile search** ejecutado desde la consola de asterisk. La dirección MAC se la puede obtener ejecutando **hcitool scan** desde la consola de Linux. El parámetro context se lo define para poder definir un plan de marcado en el archivo de configuración

extensions.conf que se encargue de las llamadas entrantes. Group es un identificador que agrupa a todos los celulares que pertenezcan a ese grupo. El parámetro nocallsetup ayuda a disminuir el retardo que se produce en ciertos dispositivos móviles al querer mostrar en pantalla el procedimiento de la llamada; con este comando le indicamos que obvie esos pasos y la ejecute directamente.

CAPITULO 4

Funcionamiento y Pruebas

Luego de la instalación de los componentes necesarios para este proyecto procedemos a realizar los pasos para conectar los gateways con asterisk, hacer llamadas desde los teléfonos IP hacia teléfonos móviles de distintas operadoras, así como también el envío de mensajes de texto desde los grandstream o simuladores de extensiones sip/iax hacia celulares.

4.1 INICIALIZANDO E INGRESANDO A ASTERISK

Los siguientes son comandos que permiten iniciar o detener el servicio de asterisk.

<i>service asterisk start</i>	Iniciar servicio.
<i>service asterisk stop</i>	Detener servicio.
<i>service asterisk status</i>	Obtener estado del servicio.
<i>service asterisk restart</i>	Reiniciar servicio.

Comandos que permiten ingresar y salir de la consola remota de asterisk así como también ejecutar acciones sin necesidad de ingresar a la misma.

<i>Asterisk</i>	Arrancar Asterisk.
<i>asterisk -c</i>	Iniciar Asterisk y abrir la consola remota
<i>asterisk -r</i>	Ingresar a la consola remota
<i>asterisk -rx 'comando'</i>	Ejecutar comando sin ingresar a la consola.
<i>stop now</i>	Detener el servicio Asterisk desde la consola remota.

<i>stop gracefully</i>	Detener el servicio Asterisk cuando todos los canales activos sean completados.
<i>stop when convenient</i>	Detener el servicio Asterisk cuando todos los canales activos sean completados. Nuevas llamadas son aceptadas. El sistema esperará hasta que no exista actividad.
<i>Exit</i>	Salir de la consola remota. No detiene el servicio Asterisk.

4.2 CONFIGURANDO EXTENSIONES SIP

Una vez que el archivo sip.conf ubicado en /etc/asterisk/sip.conf está configurado con las extensiones para uso normal (rango 1000-1899) y las extensiones privilegiadas (rango 1900-1999) procedemos a configurar los teléfonos con dichas extensiones.

Para esta implementación hemos decidido usar el teléfono grandstream con 2 líneas dadas las facilidades que brinda este dispositivo, de poder tener 2 líneas en un mismo teléfono así la línea 1 tendrá la extensión 1002 y la línea 2 será una extensión privilegiada con la numeración 1902 también haremos uso del simulador de extensiones SIP X-Lite con la extensión 1001.

4.2.1 Configurando Softphone XTEN-XLITE

Para la configuración del /Xten-xlite primero procedemos a ejecutarlo ubicándonos en la carpeta donde lo hemos descomprimido. En nuestro caso está ubicado en /xten-xlite/, nos ubicamos en esta ruta con **cd /xten-xlite/** y lo abrimos con el siguiente comando **./xtensoftphone**.

Una vez abierto el programa procedemos a realizar la configuración dando clic en el ícono *Menú* luego procedemos a dar clic en *System Settings* luego en *Sip Proxy* y por último en *[Default]*: donde configuraremos los parámetros con los siguientes datos:



Fig. 4.1 Configuración Xten-Xlite

Una vez configurado estos parámetros damos clic en back para guardar los cambios, y cerramos la ventana de menú, luego de esto el softphone intentara

autenticarse con el servidor asterisk, y de ser exitoso quedara listo para realizar las llamadas.

4.2.2 Configuración GRANDSTREAM

La configuración de este tipo de teléfono ip se lo puede realizar desde una interfaz web, para lo cual solo necesitamos de un browser que este en una pc dentro de la LAN, en este caso lo haremos desde nuestro servidor asterisk.

Primero en el menú de configuración del teléfono le asignamos una dirección ip que este dentro de nuestra red, así como una máscara de subred.

En nuestro Grandstream hemos configurado de la siguiente manera.

Ip: 192.168.1.160

Mascara: 255.255.255.0

Luego de realizar estos cambios podemos acceder mediante el browser Firefox escribiendo en la barra de direcciones la siguiente URL: <http://192.168.1.160> y damos clic en aceptar. Se mostrará una página donde debemos escribir el password el cual por defecto es *admin*, así ingresamos a administrar el teléfono.



Fig. 4.2 Administración via web - login Grandstream

A continuación se muestra la configuración de la línea 1002 (normal) del teléfono Grandstream. La interfaz web permite cambiar distintos parámetros de configuración del teléfono, pero los que son de nuestro interés y los únicos que se han modificado son los que se muestran con negrita en la figura 4.3.

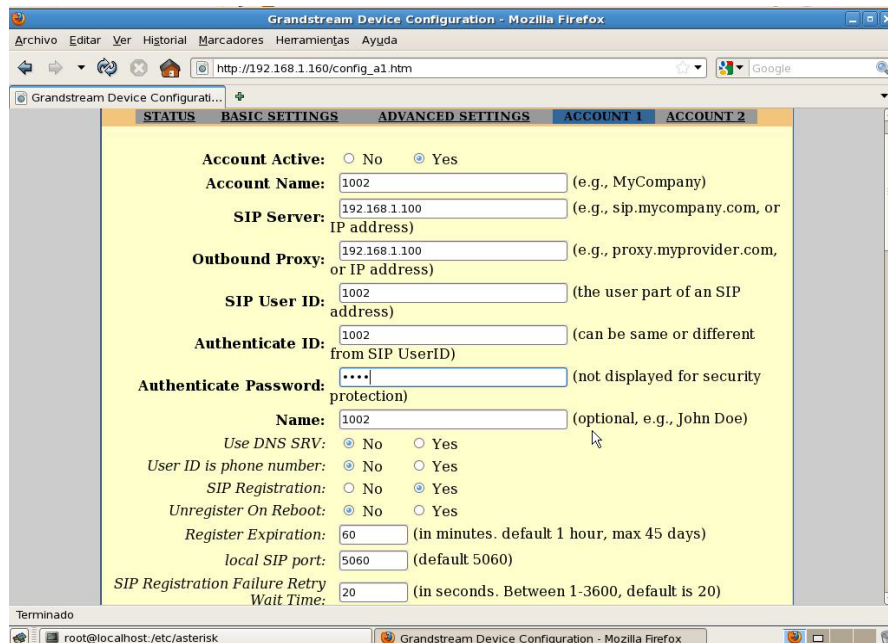


Fig. 4.3 Administración vía web - Configuración Grandstream, account 1 normal

Luego, en la siguiente línea disponible del Grandstream, se configura la extensión 1902 (Privilegiada). La interfaz web permite modificar distintos parámetros de configuración de la línea telefónica, pero los que son de nuestro interés son los que se muestran con negrita en la figura 4.4.

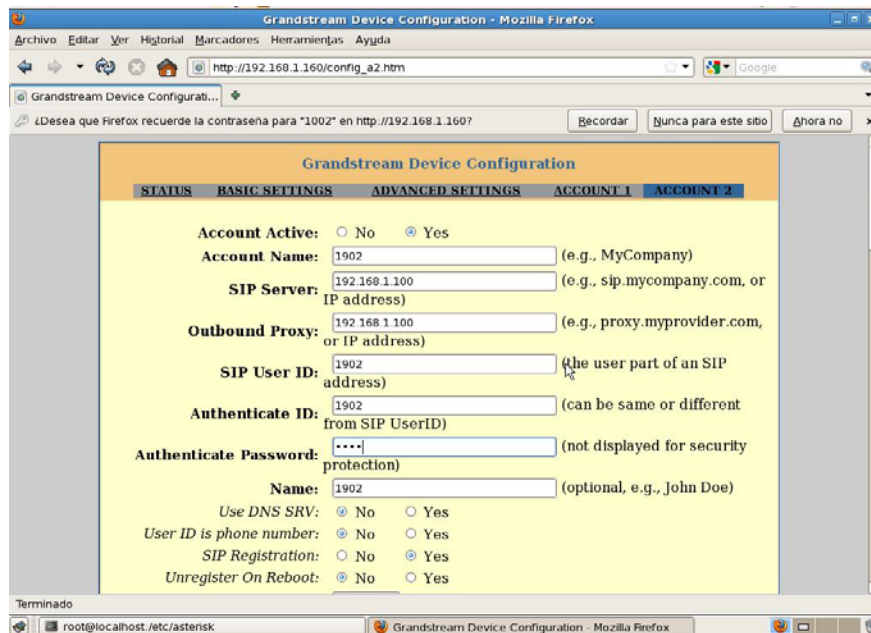


Fig. 4.4 Administración vía web-Configuración Grandstream, account 2 privilegiada

Una vez que se han ingresado estos parámetros, se debe hacer una actualización y posteriormente un reinicio del teléfono; para lo cual, primero se se da un clic en el botón update que esta al final de la página de configuración y luego aparece una página (ver Fig. 4.5) donde se presiona el botón Reboot.



Fig. 4.5 Administración vía web-Reboot Grandstream

4.3 VINCULACION DE DISPOSITIVOS MOVILES

Para vincular dispositivos móviles mediante bluetooth se necesitan de parámetros como dirección mac, puerto, también la dirección mac del bluetooth dongle que conectara los celulares con servidor asterisk; como obtenemos dichos datos se detalla a continuación.

4.3.1 Dirección MAC del Dispositivo Bluetooth Dongle

La dirección mac del dispositivo bluetooth dongle es necesaria para declararla en el archivo *mobile.conf*, dentro del contexto *adapter* el cual establecerá un canal de conexión usando este dispositivo desde los móviles que tengan asociados un adaptador con esta dirección MAC.

Para conocer la dirección MAC de los dispositivos bluetooth conectados al servidor ejecutamos el comando *hcitool dev* el cual mostrara la lista de los dispositivos conectados, como el que se muestra en la figura 4.6.

```
[root@localhost asterisk]# hcitool dev
Devices:
    hci0    00:11:67:D6:38:2E
[root@localhost asterisk]#
```

Fig. 4.6 Obtención dirección Mac dispositivo Bluetooth dongle

4.3.2 Dirección MAC y Puerto de Dispositivos Móviles

Los dispositivos móviles a conectarse con el servidor Asterisk para trabajar como Gateways también deben ser declarados en el archivo `mobile.conf` el cual se requiere la dirección MAC del celular y el número del puerto por el cual se establecerá el canal. Para obtener la dirección MAC del dispositivo se ejecuta el comando ***hcitool scan*** en la consola Shell, el cual mostrará el nombre del dispositivo y la dirección MAC asociada al mismo (ver Fig. 4.7).

```
[root@localhost asterisk]# hcitool scan
Scanning ...
    00:17:84:4E:91:FA      porta
    00:E0:03:52:99:68      movistar
```

Fig. 4.7 Obtención de dirección MAC dispositivo móvil

Para la obtención de los puertos asociados a cada dispositivo es necesario realizarlo desde la consola de asterisk. Para esto primero ingresamos a la consola remota ejecutando el comando ***asterisk -r***. Después, para mostrar los dispositivos móviles al alcance, ejecutamos ***mobile search*** y se mostrará una lista como la que se ve en la figura 4.8 donde se muestra el puerto.

```
*CLI> mobile search
Address          Name          Usable Type  Port
00:17:84:4E:91:FA porta        Yes   Phone  7
00:E0:03:52:99:68 movistar     Yes   Phone  13
```

Fig. 4.8 Obtención del puerto de dispositivo móvil

Luego de obtener los puertos de los dispositivos, se los configura en el archivo `mobile.conf`. Después se necesita actualizar dicho archivo desde la consola de asterisk. Lo hacemos ejecutando **`module unload chan_mobile.so`**, lo volvemos a cargar con **`module load chan_mobile.so`** y así se cargarán los cambios hechos. A continuación nos toca esperar que el servidor Asterisk realice una petición de conexión bluetooth al dispositivo móvil. Cuando esto suceda, ingresamos en el celular y en el servidor las mismas claves (ver Fig. 4.9) y los dispositivos estarán vinculados y listos para realizar llamadas.

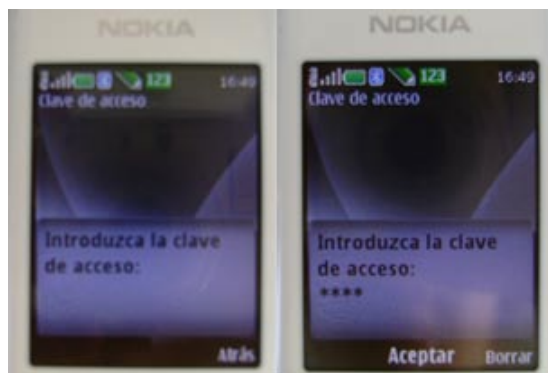


Fig. 4.9 Petición de vinculación del Servidor asterisk al dispositivo móvil

Una vez que se acoplan los dispositivos, en la consola remota de asterisk se muestran los mensajes de éxito en la conexión (ver Fig. 4.10).

```
-- Bluetooth Device movistar has connected.  
-- Bluetooth Device movistar initialised and ready.  
-- Bluetooth Device porta has connected.  
-- Bluetooth Device porta initialised and ready.
```

Fig. 4.10 Consola remota de Asterisk – vinculación exitosa

Para ver el listado de los dispositivos conectados ejecutamos el comando **mobile show devices**, el cual muestra los celulares acoplados, su estado y si soporta el envío y recepción de mensajes (ver Fig. 4.11).

```
mobile show devices
ID          Address          Group Adapter    Connected State SMS
movistar    00:E0:03:52:99:68 2    blue2           Yes     Free  Yes
porta      00:17:84:4E:91:FA 1     blue            Yes     Free  No
```

Fig. 4.11 Consola remota de Asterisk – Listado de Dispositivos acoplados

4.4 LLAMADAS A TELEFONOS CELULARES

Las llamadas se las ha realizado desde el teléfono Grandstream hacia números de líneas telefónicas de las distintas operadoras.

4.4.1 Llamadas a Porta

Como podemos apreciar en la figura 4.12 la llamada que se realiza se la hace desde el celular que esta denominado CELLG1 que es una variable en el contexto global de nuestro plan de marcado, a la cual se ha asociado el gateway Porta.

La llamada fue realizada desde la extensión 1002.

```

'CLI> == Using SIP RTP CoS mark 5
-- Executing [9094834663@internal:1] Macro("SIP/1002-09ab2c90", "outgsm_macro,Mobile/g1/094834663,CELLG1") in new stack
-- Executing [s@macro-outgsm_macro:1] Set("SIP/1002-09ab2c90", "PEXT=0") in new stack
-- Executing [s@macro-outgsm_macro:2] GotoIf("SIP/1002-09ab2c90", "0?:pext_ig0") in new stack
-- Goto (macro-outgsm_macro,s,4)
-- Executing [s@macro-outgsm_macro:4] Set("SIP/1002-09ab2c90", "INTENTOS=1") in new stack
-- Executing [s@macro-outgsm_macro:5] Set("SIP/1002-09ab2c90", "FONO=Mobile/g1/094834663") in new stack
-- Executing [s@macro-outgsm_macro:6] Set("SIP/1002-09ab2c90", "GRUPO=CELLG1") in new stack
-- Executing [s@macro-outgsm_macro:7] GotoIf("SIP/1002-09ab2c90", "0?:acc_canal") in new stack
-- Goto (macro-outgsm_macro,s,21)
-- Executing [s@macro-outgsm_macro:21] Set("SIP/1002-09ab2c90", "GROUP()=CELLG1") in new stack
-- Executing [s@macro-outgsm_macro:22] GotoIf("SIP/1002-09ab2c90", "0?:dial_pext:") in new stack
-- Executing [s@macro-outgsm_macro:23] Dial("SIP/1002-09ab2c90", "Mobile/g1/094834663,,rL{70000:10000}") in new stack
-- Limit Data for this call:
> timelimit = 70000
> play_warning = 10000
> play_to_caller = yes
> play_to_callee = no
> warning_freq = 0
> start_sound =
> warning_sound = beep
> end_sound =
-- Called g1/094834663
== Spawn extension (macro-outgsm_macro, s, 23) exited non-zero on 'SIP/1002-09ab2c90' in macro 'outgsm_macro'
== Spawn extension (internal, 9094834663, 1) exited non-zero on 'SIP/1002-09ab2c90'

```

Fig. 4.12 Consola remota de Asterisk – Llamada a Porta

En la figura 4.13 se puede observar la realización de la llamada desde el gateway de porta (el de la izquierda) hacia el celular destino.



Fig. 4.13 Dispositivos móviles – Llamada a Porta

4.4.2 Llamadas a Movistar

Como podemos apreciar en la figura 4.14 la llamada que se realiza se la hace desde el celular que esta denominado CELLG2 que es una variable en el contexto global de nuestro plan de marcado, a la cual se ha asociado el gateway Movistar.

También apreciamos que la llamada se la ha realizado desde la extensión privilegiada 1902.

```
-- Using SIP RTP CoS mark 5
-- Executing [9098172391@internal:1] Macro("SIP/1902-09d003a0", "outgsm_macro,Mobile/g2/098172391,CELLG2") in new stack
-- Executing [s@macro-outgsm_macro:1] Set("SIP/1902-09d003a0", "PEXT=0") in new stack
-- Executing [s@macro-outgsm_macro:2] GotoIf("SIP/1902-09d003a0", "17:pext_ig0") in new stack
-- Executing [s@macro-outgsm_macro:3] Set("SIP/1902-09d003a0", "PEXT=1") in new stack
-- Executing [s@macro-outgsm_macro:4] Set("SIP/1902-09d003a0", "INTENTOS=1") in new stack
-- Executing [s@macro-outgsm_macro:5] Set("SIP/1902-09d003a0", "FONO=Mobile/g2/098172391") in new stack
-- Executing [s@macro-outgsm_macro:6] Set("SIP/1902-09d003a0", "GRUPO=CELLG2") in new stack
-- Executing [s@macro-outgsm_macro:7] GotoIf("SIP/1902-09d003a0", "07:acc_canal") in new stack
-- Goto {macro-outgsm_macro,s,21}
-- Executing [s@macro-outgsm_macro:21] Set("SIP/1902-09d003a0", "GROUP()=CELLG2") in new stack
-- Executing [s@macro-outgsm_macro:22] GotoIf("SIP/1902-09d003a0", "17dial_pext:") in new stack
-- Goto {macro-outgsm_macro,s,25}
-- Executing [s@macro-outgsm_macro:25] Dial("SIP/1902-09d003a0", "Mobile/g2/098172391,,r") in new stack
-- Called g2/098172391
-- Mobile/movistar-007e answered SIP/1902-09d003a0
```

Fig. 4.14 Consola remota de Asterisk – Llamada a Movistar

En la figura 4.15 se puede observar los dispositivos participantes en la realización de la llamada, el gateway de Movistar (el de la izquierda) y el celular destinatario.



Fig. 4.15 Dispositivos móviles – Llamada a Movistar

4.4.3 Llamadas a Alegro

Como podemos apreciar en la figura 4.16 la llamada que se realiza se la hace desde el celular que esta denominado CELLG3 que es una variable en el contexto global de nuestro plan de marcado, a la cual se ha asociado el Gateway Alegro.

```
*CLI> -- Using SIP RTP CoS mark 5
-- Executing [9096344365@internal:1] Macro("SIP/1002-09ab2c90", "outgsm_macro,Mobile/g3/096141594,CELLG3") in new stack
-- Executing [s@macro-outgsm_macro:1] Set("SIP/1002-09ab2c90", "PEXT=0") in new stack
-- Executing [s@macro-outgsm_macro:2] GotoIf("SIP/1002-09ab2c90", "07:pext_ig0") in new stack
-- Goto (macro-outgsm_macro,s,4)
-- Executing [s@macro-outgsm_macro:4] Set("SIP/1002-09ab2c90", "INTENT05=1") in new stack
-- Executing [s@macro-outgsm_macro:5] Set("SIP/1002-09ab2c90", "FONO-Mobile/g3/096344365") in new stack
-- Executing [s@macro-outgsm_macro:6] Set("SIP/1002-09ab2c90", "GRUPO=CELLG3") in new stack
-- Executing [s@macro-outgsm_macro:7] GotoIf("SIP/1002-09ab2c90", "07:acc_canal") in new stack
-- Goto (macro-outgsm_macro,s,21)
-- Executing [s@macro-outgsm_macro:21] Set("SIP/1002-09ab2c90", "GROUP(=)CELLG3") in new stack
-- Executing [s@macro-outgsm_macro:22] GotoIf("SIP/1002-09ab2c90", "07dial_pext:") in new stack
-- Executing [s@macro-outgsm_macro:23] Dial("SIP/1002-09ab2c90", "Mobile/g3/096344365,,rL(70000:10000)") in new stack
-- Limit Data for this call:
> timelimit = 70000
> play_warning = 10000
> play_to_caller = yes
> play_to_callee = no
> warning_freq = 0
> start_sound =
> warning_sound = beep
> end_sound =
-- Called g3/096344365
== Spawn extension (macro-outgsm_macro, s, 23) exited non-zero on 'SIP/1002-09ab2c90' in macro 'outgsm_macro'
== Spawn extension (internal, 9096344365, 1) exited non-zero on 'SIP/1002-09ab2c90'
```

Fig. 4.16 Consola remota de Asterisk – Llamada a Alegro

También apreciamos que la llamada se la ha realizado desde la extensión 1002. En la figura 4.17 se observa a los celulares participantes en la llamada: el gateway de alegre (el de la izquierda) y el celular destinatario.



Fig. 4.17 Dispositivos móviles – Llamada a Alegre

4.5 ENVIO DE MENSAJES DE TEXTO A TELEFONOS CELULARES

El envío de mensajes se lo ha procedido a realizar desde el teléfono Grandstream. Dadas las compatibilidades analizadas en el capítulo 3, ya que solo el modelo **Nokia 6230** que se está utilizando como gateway de movistar es capaz de enviar mensajes de texto se ha considerado este celular para que realice el envío a los 3 posibles destinos (celulares de la telefonía porta, movistar, alegre).

A continuación se mostrara cada uno de los mensajes definidos como plantillas (ver tabla V)

4.5.1 Envío de Mensaje Plantilla 1 (Ver Tabla VI)

Como se puede apreciar en la figura 4.18, el número que se marco fue el 00109483466324091200, el mismo que está compuesto por:

0	extensión mensajes de texto
01	numero de plantilla
094834663	destinatario
2409	fecha
1200	hora

Y el mensaje que recibió el destinatario fue ***“La Reunión está establecida para el 24/09 a las 12:00”***



Fig. 4.18 Dispositivos móviles – Envío de mensaje plantilla 1 (ver tabla VI)

4.5.2 Envío de Mensaje Plantilla 2 (Ver Tabla VII)

Como se puede apreciar en la figura 4.19, el número que se marcó fue el 002094834663, el cual está compuesto por:

0	extensión mensajes de texto
02	numero de plantilla
094834663	destinatario

Y el mensaje que recibió el destinatario fue ***“La Reunión se ha cancelado.”***



Fig. 4.19 Dispositivos móviles – Envío de mensaje plantilla 2 (ver tabla VII)

4.5.3 Envío de Mensaje Plantilla 3 (Ver Tabla VIII)

Como se puede apreciar en la gráfica de la figura 4.20, el número que se marcó fue el 00309483466330091420, el cual está compuesto por:

0	extensión mensajes de texto
03	numero de plantilla
094834663	destinatario
3009	fecha
30101420	hora

Y el mensaje que recibió el destinatario fue ***“No me encuentro disponible, Llameme el 30/09 a las 14:20”***



Fig. 4.20 Dispositivos móviles – Envío de mensaje plantilla 3 (ver tabla VIII)

4.5.4 Envío de Mensaje Plantilla 4 (Ver Tabla IX)

Como se puede apreciar en la grafica de la figura 4.21, el número que se marcó fue el 004098172391094834663, el mismo que está compuesto por

0	extensión mensajes de texto
04	numero de plantilla
098172391	destinatario
094834663	numero celular

Y el mensaje que recibió el destinatario fue **“Por Favor llame al 094834663”**



Fig. 4.21 Dispositivos móviles – Envío de mensaje plantilla 4 (ver tabla IX)

4.5.5 Envío de Mensaje Plantilla 5 (Ver Tabla X)

Como se puede apreciar en la figura 4.22, el número que se marcó fue el 0050981723911530, el mismo que está compuesto por

0	extensión mensajes de texto
05	numero de plantilla
098172391	destinatario
1530	hora

Y el mensaje que recibió el destinatario fue “**Llego tarde, Estaré ahí a las 15:30’**”



Fig. 4.22 Dispositivos móviles – Envío de mensaje plantilla 5 (ver tabla X)

CONCLUSIONES Y RECOMENDACIONES

Conclusiones

- 1) El software libre provee de herramientas útiles y adaptables al entorno, gracias a su soporte y su universo de que se encuentran en continuo crecimiento y mejoramiento, sin costo alguno.
- 2) El uso de bluetooth para conexión a red GSM es más económico que adquirir una tarjeta FXO/FXS para conectar una base celular.
- 3) La redirección de llamadas según la operadora destino ahorra costos de interconexión.
- 4) La concurrencia de llamadas manejada eficazmente, aprovechará de mejor manera los recursos.
- 5) Toda la gama de opciones que brinda el software libre nos libra de las limitaciones del software privativo, en el cual solo estamos destinados a ser meros espectadores. Esto nos ayuda a encontrar cualquier tipo de solución de igual manera que si pagáramos por algún sistema, pero con la libertad de mejorar los resultados en nuestro requerimiento y a mucho menor costo.
- 6) Con la ley de portabilidad numérica, los números por sí solos no indicarían a que operadora pertenecen (Ver ANEXO A), por lo que nuestro plan de mercado dejará de funcionar y las llamadas podrían salir por la operadora que no corresponde.

Recomendaciones

- 1) Determinar las características de los equipos servidores en función del número de usuarios y tipo de servicios.
- 2) Contratar el Ancho de Banda necesario para el buen funcionamiento del sistema dependiendo de las funcionalidades y el número de usuarios.
- 3) Tener precaución de que los dispositivos móviles que se comunican al servidor y los dispositivos bluetooth, se encuentren dentro de la lista de equipos que funcionan correctamente, para evitar malos resultados en la calidad del servicio.
- 4) Hasta que se implemente la portabilidad hay que tener cuidado de siempre actualizar los códigos de las operadoras ya que aparecen nuevos códigos que si no los actualizamos podríamos llamar por otra operadora que no corresponda.
- 5) Se recomienda el siguiente trabajo a futuro para solucionar el problema de la portabilidad numérica.

MÉTODO ALL CALL QUERY



Interacción de Operadoras

Utilizando el método de All Call Query (ver ANEXO A), se generará diariamente el archivo de números portados y los números a ser portados al día siguiente. Se establecerá una ventana de cambio a ser utilizada por los concesionarios de servicios que será de las 00:00 a las 06:00 horas, el periodo máximo de interrupción de servicio para el usuario será de 3 horas.

De esta manera habrá que obtener a qué empresa se deberá llamar y solucionar el problema de ya no poder crear un dial plan por código.

GLOSARIO DE TERMINOS

Add-on: Pequeños programas opcionales que sólo funcionan anexados a otro y que sirven para incrementar o complementar sus funcionalidades.

Bluetooth: Nombre común de la especificación industrial **IEEE 802.15.1**, que define un estándar global de comunicación inalámbrica y posibilita la transmisión de voz y datos entre diferentes dispositivos mediante un enlace por radiofrecuencia segura, globalmente y sin licencia de corto rango.

Dialplan: Un dial plan (plan de marcado) es un software residente en algunos gateways y en casi todas las IP-PBX que permite determinar el tratamiento que debe darse a un número discado.

Gateway: Puerta de enlace, acceso, pasarela. Nodo en una red informática que sirve de punto de acceso a otra red.

GSM: Son las siglas de Global System for Mobile communications (Sistema Global para las comunicaciones Móviles), es el sistema de teléfono móvil digital más utilizado y el estándar de facto para teléfonos móviles en Europa.

MAC: Media Access Control (Control de Acceso al Medio). Es un número de serie puesto en el hardware de los adaptadores de red y sirve para identificarlos

de manera única de otras tarjetas de red. Está compuesto por 48 bits, los primeros 24 identifican a la empresa fabricante y el resto un número único.

Macro: Una macro o macroinstrucción es una serie de datos que se almacenan para que se puedan ejecutar de forma secuencial mediante una sola llamada u orden de ejecución.

Open Source: es el término con el que se conoce al software distribuido y desarrollado libremente. El código abierto tiene un punto de vista más orientado a los beneficios prácticos de compartir el código que a las cuestiones morales y/o filosóficas las cuales destacan en el llamado software libre.

PBX: (Central Telefónica Digital). Sistema telefónico dentro de una organización que maneja las llamadas entre sus usuarios en líneas locales mientras permite que entre todos los usuarios compartan un número determinado de líneas telefónicas externas.

Software Propietario: Se refiere a cualquier programa informático en el que los usuarios tienen limitadas las posibilidades de usarlo, modificarlo o redistribuirlo (con o sin modificaciones), o cuyo código fuente no está disponible o el acceso a éste se encuentra restringido.

Voip: Voice Over Internet Protocol (Voz Sobre el Protocolo de Internet). Básicamente VoIP es un método por el cual tomando señales de audio analógicas del tipo de las que se escuchan cuando uno habla por teléfono se las transforma en datos digitales que pueden ser transmitidos a través de internet hacia una dirección IP determinada.

ANEXOS

ANEXO A

PORTABILIDAD NUMERICA

Dispone que todo abonado de los servicios de telecomunicaciones móviles tiene el derecho a mantener su número telefónico móvil, un cuando cambie de red, servicio o empresa operadora. **(Ref. Mandato Constituyente No 10).**

La portabilidad del número permitirá a un abonado mantenerlo aun cuando cambie de prestador o domicilio. **(Ref. Plan Técnico Fundamental Nacional vigente).**

IMPLEMENTACION DE PORTABILIDAD NUMERICA

Entre las formas de implementar se encuentra las siguientes:

- **Onward Routing** – Encaminamiento hacia adelante
- **Call Drop Back** – Devolución de llamada
- **Query on Release** - Consulta para liberación.
- **All Call Query** – Consulta de todas las llamadas.

Por considerarse una solución técnicamente superior, All Call Query ha sido adoptada en la mayoría de las soluciones para portabilidad numérica.

FUNCIONAMIENTO DE ALL CALL QUERY

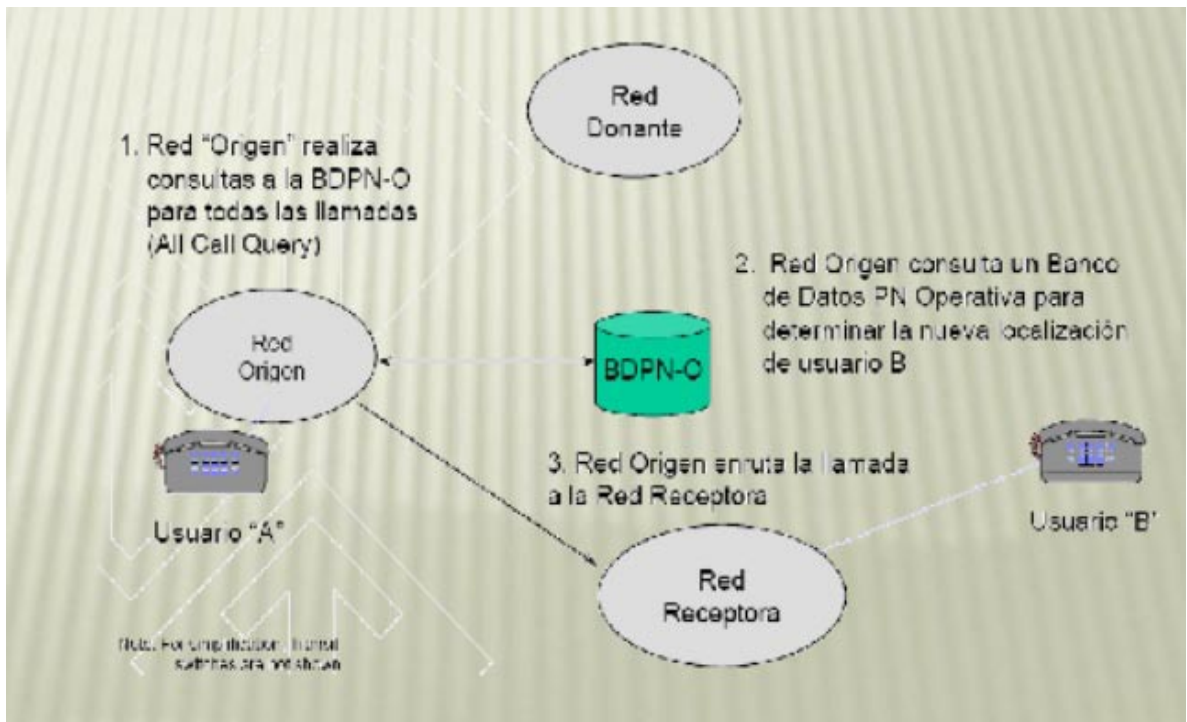


Fig. A-1 Arquitectura de All Call Query

La red de origen consulta a una base de datos central donde se almacena la información para poder encontrar al usuario destino (ver Fig. A-1).

Todo esto lo logra con la colaboración de la Red Donante y la Red Receptora.

BILIOGRAFIA

[1] Van Meggelen J., Smith J. y Madsen L., Asterisk The Future of Telephony, Editorial O'Reilly Media, 2005.

[2] Umanzor P, Conectar Asterisk a la red GSM - chan mobile ,
http://info.linuxmall.cl/wiki/index.php/Conectar_Asterisk_a_la_red_GSM_-_chan_mobile, 2009

[3] Voip IP Telephony, Configuring, using and debugging chan_mobile ,
<http://snapvoip.blogspot.com/2007/10/configuring-using-and-debugging.html>, 2007

[4] Voip-Info.org , Chan mobile, http://www.voip-info.org/wiki/view/chan_mobile, 2008

[5] Talking Around the world, Instalación de Asterisk 1.6.x en Ubuntu 8.1,
<http://www.voztovoice.org/?q=node/165>, 2009

[6] Linux Mall, ANX/IP--ASTERISK--BLT/Dongle—GSM ,
<http://bytecoders.homelinux.com/files/ast-gsmv3.pdf>, 2007