

Técnicas de diseño de pasarelas de VoIP empleando estándares abiertos y código libre

Por Ing. Juan Alain Capote Álvarez, Profesor Instructor, e Ing. Alian Manuel Capote Álvarez, Profesor Instructor, Cátedra de Comunicaciones, Instituto Técnico Militar José Martí, ITM.
manuelcapote@infomed.sld.cu

I Introducción

Desde la invención del teléfono a finales del siglo XIX, las compañías telefónicas se han orientado de manera casi exclusiva a las comunicaciones de voz. Durante muchos años la industria ha realizado continuos cambios encaminados a mejorar la calidad, fiabilidad y costos de los sistemas telefónicos, introduciendo la conmutación digital, la transmisión por fibra óptica, las redes inteligentes y miles de facilidades en el tratamiento de las llamadas; sin embargo, el carácter de las redes telefónicas por conmutación de circuitos ha permanecido sin variaciones, y se ha identificado por un alto consumo de recursos y costos elevados [1].

El surgimiento de las redes de conmutación de paquetes brindó una nueva infraestructura para el advenimiento de una tecnología capaz de transmitir voz sobre las redes de datos. En el futuro cuando los datos sean el componente dominante de la demanda de tráfico,

será más práctico cursar la voz por este tipo de redes en lugar de forzar a los datos para que se ajusten a la red telefónica y, ciertamente, no es económico continuar instalando, operando y administrando redes separadas [2].

En Cuba la infraestructura tecnológica continuará el mejoramiento de las redes de comunicaciones, a la par del incremento y modernización del parque de equipamiento. En este sentido, el país ha organizado el uso racional y eficiente de los recursos tanto de equipamiento como de conectividad.

Este desarrollo ocasionará que las Tecnologías de la Información y las Comunicaciones (TIC) tiendan hacia el entorno que proponen las Redes de Próxima Generación —del inglés, *Next Generation Networks*—. No obstante, la infraestructura de telecomunicaciones no puede ser sustituida radicalmente, debido a los elevados costos que satisfacen esta condición. Por

lo tanto existirá, al menos temporalmente, una fase de coexistencia y, por supuesto, la necesaria interconexión mediante pasarelas como un elemento imprescindible para la migración.

Este trabajo centra sus objetivos en crear un documento breve y pedagógico con la idea de proponer una serie de pasos para diseñar y configurar un sistema que permita la comunicación bidireccional con la Red Telefónica Pública Conmutada (RTPC) y su interoperabilidad con la gran variedad de normas de señalización de llamada utilizando la plataforma telefónica de código abierto Asterisk. De esta forma, posibilita que, en nuestro país, se incrementen progresivamente los niveles de seguridad y tienda indiscutiblemente hacia la necesaria independencia tecnológica.

2 Pasarelas de medios

Una pasarela es un dispositivo que permite interconectar redes con pro-

protocolos y arquitecturas diferentes a todos los niveles de comunicación. Su propósito es traducir la información del protocolo utilizado en una red, al protocolo empleado en la red de destino. Una pasarela de medios puede funcionar como [3]:

- ♦ Pasarela de Acceso —del inglés, *Access Media Gateway* (AMG)—: proporciona acceso a líneas telefónicas analógicas, Red Digital de Servicios Integrados (RDSI) y a líneas digitales de abonado. El AMG realiza labores de compresión y descompresión de señales de voz, por lo que requiere potencia de procesamiento.

- ♦ Pasarela de Entronque —del inglés, *Trunking Media Gateway* (TMG)—: es una pasarela entre una red con conmutación de circuitos y una red de paquetes. Las TMG se despliegan en el borde de la red de paquetes, cerca del ingreso de las líneas troncales.

- ♦ Pasarela de Señalización —del inglés, *Signaling Media Gateway* (SMG)—: funciona como una interfaz de señalización con la PSTN —*Public Switched Telephone Network*—. La SMG termina las conexiones de Canal Común Número 7 (SS7) y emula un terminal para la red SS7.

- ♦ Pasarela de VoIP: dispositivo o programa capaz de interconectar los distintos estándares de señalización de llamada de la tecnología VoIP.

- ♦ Pasarela Universal de Medios —del inglés, *Universal Media Gateway* (UMG)—: convierte flujos de medios y señalización en flujos IP —*Internet Protocol* / *Protocolo de Internet*—. Una pasarela universal puede actuar como conmutador TDM, TMG, SMG y AMG.

2.1 VoIP, estándares abiertos y código libre

Una definición general de Voz sobre IP es la posibilidad de transportar conversaciones telefónicas en paquetes IP. El término VoIP no se refiere a ninguno de los mecanismos concretos que existen para trasladar las señales de voz, este se fundamenta en la conmutación

de paquetes en vez de los métodos tradicionales de telefonía, basada en la conmutación de circuitos. La voz sobre IP convierte las señales de voz estándar en paquetes de datos comprimidos que son transportados a través de redes de datos [4].

Existen docenas de tecnologías que permiten la comunicación de voz a través de la red, estas se pueden dividir en dos grandes grupos: tecnologías cerradas-propietarias y sistemas abiertos. En el primer grupo se encuentran el conocido Skype o el ya legendario SCCP —*Skinny Client Control Protocol* / *Protocolo de Control de Clientes Ligeros*— de Cisco; en el segundo grupo, los estándares abiertos basados en SIP —*Session Initiation Protocol* / *Protocolo de Inicio de Sesión*—, H.323 o IAX —*Inter-Asterisk eXchange Protocol* / *Protocolo de Intercambio entre sistemas Asterisk*—.

Los sistemas código libre se caracterizan por contar con software que se distribuye junto con su código fuente y con una licencia de uso. Esto permite que sean instalados, configurados y actualizados sin necesidad de la intervención del que los suministra. También, facilitan la programación de nuevos servicios que pueden someterse a prueba por un amplio número de especialistas. Con el código libre puede aprenderse de experiencias parecidas, integrar sus soluciones y compartir resultados con los demás.

Una de las primeras preguntas que merece una respuesta es: ¿por qué debería crearse una infraestructura propia de voz sobre IP y no seguir usando los servicios ya existentes?

La respuesta es simple: sostenibilidad y flexibilidad. Los servicios existentes pueden solucionar una necesidad a corto plazo, pero nunca garantizar la independencia o el control del propio proceso de aprendizaje y desarrollo. No se trata de una cuestión puramente técnica. El pro-

blema no es decidir cuál es la mejor de las tecnologías; sino, cuál es la que permite que las comunidades sean dueñas de su propio desarrollo y que puedan adaptarla a sus propias necesidades. Es difícil imaginar un desarrollo sostenible sin transferencia de conocimiento y reapropiación tecnológica. Una solución basada en estándares abiertos y código libre no es sólo una buena solución desde un punto de vista puramente técnico, sino que además permite la posibilidad de adaptación para mejorarse a la realidad local.

2.2 Asterisk el futuro de la telefonía

En la actualidad los sistemas de comunicaciones permiten integrar cada vez más tecnologías diferentes y, de este modo, posibilita que usuarios que utilizan otras tecnologías como conmutación de paquetes y de circuitos, puedan comunicarse de forma transparente. Desafortunadamente, este equipamiento aún es muy caro para las empresas nacionales, por lo que se necesitan buscar soluciones que permitan la integración de nuestra infraestructura de comunicaciones a un costo mucho menor. Dicha integración también proporcionará el uso de los beneficios que brinda el protocolo IP. Las soluciones que emplean software de código abierto, como es el caso de Asterisk, son de las más atractivas para permitir la integración.

Asterisk es una central telefónica digital en software bajo licencia pública general que integra las funcionalidades de telefonía clásica con nuevas capacidades derivadas de su flexible y potente arquitectura. Funciona como una aplicación “servidor” que permite que terminales “clientes” se conecten a él. Una vez conectados, los usuarios pueden transmitir voz y video en tiempo real utilizando cualquiera de los protocolos y *codecs* sustentados por esta central.

Al soportar una mezcla de la telefonía tradicional y los servicios de VoIP, Asterisk permite construir nuevos sistemas telefónicos eficien-

tes o migrar gradualmente a los sistemas existentes de las nuevas tecnologías [5]. La forma modular es lo que le permite a Asterisk integrar los hardwares de telefonía implementados y la tecnología de VoIP.

Para facilitar la abstracción de protocolos y de hardware, se determinan 4 módulos que tienen la capacidad de cargar 4 interfaces de programas de aplicación que son definidas alrededor del núcleo del sistema y proveen el uso de aplicaciones a través de módulos cargables para realizar cualquier acción en demanda. También permite un desarrollo abierto de nuevas aplicaciones para satisfacer necesidades o situaciones únicas [5].

El código del programa fue originalmente creado por Mark Spencer basado en las ideas y el trabajo previo de Jim Dixon —proyecto de telefonía Zapata—. El programa, sus mejoras y correcciones, es el resultado del trabajo colectivo de la comunidad del software libre. Sin embargo, Digium es el principal patrocinador de Asterisk; y Mark Spencer, su creador y soporte principal. Para usar Asterisk sólo se necesita una Computadora Personal; pero, en caso de querer conectarse a la PSTN, es necesario añadir un hardware especializado.

3 Metodología para el diseño

A continuación se presentan una serie de pasos para la elaboración de una pasarela de VoIP con acceso troncal a la PSTN.

3.1 Selección del servidor

La selección de un servidor para que Asterisk corra en él es sencilla y, al mismo tiempo, compleja; fácil porque cualquier máquina fabricada después del año 2000 debe tener suficiente potencia para hacerlo funcionar. Sin embargo, a medida que el sistema crezca —especialmente si se emplean *codecs* de alta compresión— habrá que considerar el mejoramiento del procesador y la memoria. Cuando se

selecciona el servidor hay que valorar, cuidadosamente, el diseño del sistema y las funcionalidades que se desean cubrir [5].

El tamaño de un sistema basado en Asterisk no se determina por el número total de usuarios al que brindará servicios; sino, por la cantidad de llamadas simultáneas que se espera que soporte, ya sean internas de VoIP o hacia la PSTN. Esto se debe a que Asterisk usa el CPU —*Central Processing Unit* / Unidad Central de Procesamiento— del servidor para procesar los canales de voz, en vez de tener un DSP —*Digital Signal Processor* / Procesador Digital de Señales— dedicado a cada canal, haciendo que el rendimiento del sistema sea muy dependiente del rendimiento del CPU. Por lo tanto, se precisa conocer la cantidad de llamadas simultáneas máximas que va a tener el sistema. En la tabla I se recomiendan los requerimientos mínimos del servidor por cantidad de llamadas simultáneas.

Propósito	Llamadas simultáneas	Hardware mínimo recomendado
Sistemas de prueba	no más de 5	400 MHz 256 MB RAM
Pequeñas empresas	de 5 a 10	900 MHz 256 MB RAM
Medianas empresas	hasta 25	3 GHz 512 MB RAM
Sistemas a gran escala	de 25 en adelante	procesadores <i>dual core</i> o servidores en una arquitectura distribuida

Tabla 1 Requerimientos mínimos recomendados para el servidor [5].

Como se observa en la tabla, para sistemas grandes de más de 25 llamadas simultáneas pudieran utilizarse servidores en una arquitectura distribuida, donde cada servidor se encargaría de atender una cantidad de usuarios específica y entre estos se enlazarían a través del protocolo IAX, lo que permitirá un aprovechamiento del ancho de banda de la red. Además, se pueden emplear servidores de apoyo que se encarguen de los servicios avanzados tales como: buzón de voz, conferencia, base de datos, etc. Esto posibilita que los servidores centrales sólo se encarguen del procesamiento de la llamada, lo cual mejora la Calidad del Servicio —del inglés, *Quality of Service* (QoS)—.

El hardware necesario para Asterisk no es muy complicado, no se precisa de una tarjeta de video sofisticada o periféricos. Puertos seriales, paralelos y USB pueden ser completamente deshabilitados; exclusivamente una buena tarjeta de red es esencial, además de dejar libres al menos dos *slots* PCI y conectores de alimentación *molex* —5v / 12v— para las tarjetas de Digium que permiten la conexión a la red de telefonía pública.

3.2 Consideraciones del Sistema Operativo e instalación de Asterisk

Aunque Asterisk puede funcionar en muchos Sistemas Operativos (SO), GNU/Linux es la plataforma más estable y en la que existe un mayor soporte. Al seleccionar un SO, deben tenerse en cuenta las mismas consideraciones que al escoger un hardware, y se le añaden facilidades de uso, gestión, administración, seguridad, etc. Entre las distribuciones Linux más utilizadas se recomiendan: Ubuntu, Debian, Red Hat y Centos.

En estos momentos se ha creado en la Universidad de Ciencias Informáticas (UCI) del país, la plataforma de desarrollo personalizada Nova LNX, la cual no es más que un ensayo, en menor escala, de un proceso que es inevitable a nivel nacional. La necesaria integración y el

constante desarrollo de Asterisk junto con Nova LNX, permitirá la creación de un sistema meramente cubano, capaz de competir con las distribuciones ya existentes que utilizan Asterisk como una solución de telefonía

Asterisk no es sólo un programa con un diseño y arquitectura potente y flexible de gran tamaño; sino que, además, sigue creciendo al integrar más funcionalidades cada día. La flexibilidad de Asterisk también implica cierta complejidad debido a las numerosas posibilidades y opciones.

El paquete de instalación de Asterisk necesita algunas dependencias para poder compilar: inicialmente se debe instalar el compilador Gcc —Versión 3.x o superior— y sus dependencias. También requiere el paquete Bison y los paquetes de desarrollo Ncurses para la funcionalidad de la Interfaz de Comando de Línea (CLI). La biblioteca criptográfica en Asterisk exige OpenSSL y sus paquetes de desarrollo. Al utilizarse hardware de Digium, para que algunas aplicaciones sean incluidas en tiempo de compilación, es necesario instalar el paquete Zaptel, el cual requiere Libnewt y sus dependencias. Las bibliotecas Zlib y Zlib-devel son necesarias para compilar.

Asterisk es compilado con Gcc a través del uso del programa Make. A diferencia de muchos otros programas, no hay necesidad de ejecutar un *script* de configuración para su instalación. Para instalar Asterisk simplemente se ejecutan los comandos que se muestran a continuación:

```
# make menuselect
# make install
# make samples
```

El comando **make menuselect** posibilita una selección más detallada de los módulos que se desean instalar antes de compilar el software y, de esta forma, no se requiere que el usuario edite los archivos de instalación.

Para agregar soporte a las tarjetas que permiten la conexión a la PSTN

es necesario compilar e instalar el controlador Zaptel, para esto hay que actualizar las cabeceras del kernel de Linux instalando el paquete Linux-headers-2.6.xx-xx-386.

3.3 Configurando Asterisk

Para configurar Asterisk de forma tal que funcione como una pasarela multiprotocolo y lograr la interoperabilidad entre los distintos estándares de VoIP, es preciso editar los siguientes archivos de configuración basados en texto: sip.conf, h323.conf, skinny.conf e iax.conf.; además del archivo extensions.conf que es el responsable del plan de marcado dentro del servidor Asterisk.

La configuración de los “clientes” en Asterisk es muy similar entre los diferentes estándares abiertos, donde cada usuario es identificado por un contexto, el cual es el nombre de usuario asociado a este o es un nombre arbitrario utilizado por otros archivos de configuración para referirse a este dispositivo. Por ejemplo, un cliente SIP se configuraría de la siguiente forma:

```
[1000]
type=friend
context= sip
callerid= xlite <1000>
secret=1000
host=dynamic
```

Como se percibe en el fragmento anterior, se definió un usuario SIP que se identificará con el número 1000 y se utilizará el tipo de usuario Friend para realizar y recibir llamadas. El parámetro “context”, indica a qué contexto del plan de marcado pertenece este dispositivo y, en dependencia de las inclusiones realizadas en este, podrá efectuar o no llamadas a otros usuarios definidos en otros contextos.

El “callerid” no es más que la identificación de la persona que está realizando la llamada, o sea, el nombre que se muestra a otros clientes cuando reciben una llamada de este dispositivo. Para la autenticación con el servidor, se utiliza el parámetro “secret” que será la contraseña del cliente SIP. Esto junto con el parámetro “host” que indica la dirección IP del usuario, sirve para una identificación completa del usuario, al evitar la suplantación de identidad. El parámetro “host” se puede configurar con la opción “dynamic” para permitir que el usuario pueda conectarse al servidor desde cualquier punto de la red.

En el caso de los clientes de protocolos propietarios como el SCCP, la configuración es un poco compleja, debido a que los dispositivos Cisco son teléfonos Ethernet (*Ephones*). Cada *Ephone* puede tener múltiples extensiones asociadas a él y un único número de secuencia para identificarlo durante la configuración, este identificador se basa en las siglas SEP —del inglés, *Selsius Ethernet Phone* / Teléfono Ethernet de Selsius— seguido por la dirección MAC —*Media Access Control* / Control de Acceso al Medio— del dispositivo. En caso de ser un *softphone*, esta sería la dirección MAC de la tarjeta de red de la PC donde está corriendo. Asterisk define esta identificación a través del parámetro “device”. Para añadir las extensiones pertenecientes a un dispositivo se utiliza “line”, a este parámetro se le coloca el valor del número telefónico que tendrá el dispositivo. El parámetro “context” puntualiza a qué contexto del plan de marcado pertenecen el dispositivo y las extensiones.

```
[devices]
device=SEP00D0C9582946
host=192.168.0.1
context = sccp
callerid="Alain Capote" <4000>
line => 4000
```

3.4 Integrar VoIP con la telefonía tradicional

Básicamente existen dos opciones de líneas telefónicas hoy en nuestro país. Las líneas analógicas que son las más comunes y son entregadas usando un par metálico de hilos de cobre; y las líneas digitales, empleadas cuando son necesarias muchas líneas analógicas.

El término FX —*Foreign Xchange* / Intercambio Foráneo— es aplicado para troncos con acceso a la PSTN. Las interfaces FXO —*Foreign Xchange Office* / Oficina de Intercambio Foráneo— se utilizan básicamente para la comunicación de un puerto de Asterisk con la PSTN. Esta comunicación requiere tono de marcado, indicación de timbre, indicadores de llamada en progreso y otros. Las interfaces FXO conectan Asterisk a otro conmutador PBX —*Private Branch Exchange* / Central Telefónica Privada—, a la PSTN o a una pasarela de VoIP. Las interfaces FXS —*Foreign eXchange Station* / Estaciones de Intercambio Foráneo— pueden ser utilizadas para conectar dispositivos básicos tales como: teléfonos, módems y faxes. Deben proveer voltaje, generar tono de timbre, detección de descolgado e indicar llamadas en progreso [6].

Es necesario el uso de hardware adicional para que Asterisk pueda funcionar como una pasarela de entronques, tales como las tarjetas analógicas o digitales de Digium.

Las tarjetas analógicas son como módems especialmente dedicadas a gestionar el tráfico de voz. Estas son necesarias para la conexión de líneas troncales y teléfonos analógicos, a través del empleo de módulos FXS o FXO, los cuales se distribuyen con o separados de la tarjeta. Para cada tarjeta es preciso editar el archivo `zaptel.conf` y `zapata.conf` que son los encargados de la configuración del dispositivo. La empresa Digium las distribuye con 4, 8 y 24 puertos que pueden ser: FXS, FXO o una combinación de estos [5].

Hasta el 2007 las más utilizadas fueron las del modelo TDM400P, las cuales no se están comercializando actualmente, debido a que fueron reemplazadas por la serie TDM410P (Figura 1) que presenta una serie de ventajas frente a esta. A este grupo de tarjetas se le proporcionó la nomenclatura TDMXYZ de la siguiente forma:

- ♦ X: número de interfaces FXS que posee

- ♦ Y: número de interfaces FXO que posee

- ♦ Z: puede tomar dos valores: E para indicar que se incluirá en la tarjeta un módulo de cancelación de eco o B que no se incluirá.

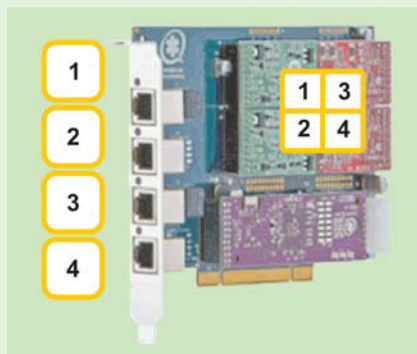


Figura 1 Tarjeta analógica TDM410P de Digium. (Fuente: elaboración propia).

Es importante considerar la forma en que se colocan los módulos, porque si por error se conectara un puerto FXS a la PSTN, esto podría causar daños en el módulo y a la tarjeta. El primer puerto siempre se encuentra en el lado más alejado del bus PCI de la PC y los módulos se insertan en orden (Figura 1).

Las tarjetas digitales implementadas por Digium permiten la conexión de troncos digitales E1 (32 canales), T1 (24 canales) y flujos primarios RDSI. Hay que prestar atención al escoger una tarjeta debido a que algunos modelos como la TE405P utilizan un puerto PCI de 5v. Este tipo de puerto raramente es encontrado en las PC comunes, mientras que la TE410P y la TE212P funcionan con puertos PCI de 3,3v, disponibles habitualmente

en la mayoría de las placas base modernas y en arquitecturas de puertos de 64 bits.

3.4.1 Cálculo de tráfico telefónico

Aunque es teóricamente posible suministrar servicios de telefonía para que todos los usuarios configurados en Asterisk puedan comunicarse simultáneamente con la PSTN, la probabilidad de que esta situación ocurra es muy baja. De manera que el enorme gasto que satisface esta condición no es justificable.

En la práctica, aún durante los periodos más ocupados del día, sólo una pequeña porción de los abonados demandará servicio telefónico. Por lo tanto, en estos casos es útil realizar un análisis estadístico exhaustivo para conocer la cantidad máxima de llamadas hacia la PSTN que procesará el sistema en la hora más activa del día, y obtener un promedio de duración de estas llamadas. Con estos datos y fijando un buen grado de servicio, se aplica la ecuación (1) y se obtiene el tráfico telefónico en Erlang. Con este otro dato, utilizando las tablas o una calculadora de Erlang B, puede obtenerse cuántas líneas troncales serán necesarias para satisfacer dicho tráfico telefónico.

$$A = \frac{C \cdot tr}{T} = \frac{15 \cdot 4}{60} \text{ Erl} \quad [1]$$

donde:

A – Intensidad de tráfico

C – Número promedio de llamadas

tr – Tiempo promedio de duración de las llamadas

T – Tiempo de observación

En este punto en dependencia de la cantidad de líneas que sean necesarias, sería útil repasar nuevamente el paso #1, recordando siempre que, al utilizar servidores en una arquitectura distribuida, el sistema se desempeña de una manera más eficiente y se distribuyen las tareas, y esto trae como consecuencia el mejoramiento de la Calidad del Servicio.

3.5 Configurando Zaptel para las tarjetas analógicas

Para cada tarjeta es preciso configurar el archivo `zaptel.conf` que, al contrario de los demás archivos de configuración, se encuentra en el directorio `/etc`. A continuación se muestra un segmento de este archivo donde se agregan dos puertos FXS con señalización FXO y un puerto FXO con señalización FXS.

```
fxoks= 1
fxoks= 2
fxsks= 4
loadzone=cu
```

El “loadzone” define el *set* de indicaciones a usar por el canal y se configura en el archivo `zonedata.c`. Este contiene información acerca de todos los sonidos que un sistema telefónico hace en un país particular, entre ellos: el tono de marcar, ciclos de timbre, la señal de ocupado y otros. Los parámetros “fxsks” y “fxoks” especifican el protocolo de señalización que van a utilizar las interfaces FX; existen tres opciones [5]:

- ♦ Loop start (ls)
- ♦ Ground start (gs)
- ♦ Kool start (ks)

La diferencia entre los protocolos de señalización Loop start y Ground start viene dada por la forma en que el equipo solicita el tono de marcado. Un circuito Ground start hace señales momentáneamente al extremo que requiere tono de marcado, y pone en tierra una de las líneas. Un circuito Loop start utiliza un cortocircuito para solicitar tono de marcado. Kool start es de hecho lo mismo que Loop start, sólo que tiene más habilidad para descubrir cuándo hay cortocircuito o desconexión en el extremo. Para configurar un método de señalización aparte de Kool start, se reemplaza los “ks” con “ls” o “gs” —para Loop start o Ground start, respectivamente—.

Luego se debe verificar el estado de los canales creados utilizando el comando “`ztcfg`”, el cual irá a examinar la configuración en el archivo `zaptel.conf`. Se puede comprobar que el hardware y puertos

se cargaron y configuraron correctamente con el uso de la línea de comando:

```
# /sbin/ztcfg -vv
```

Para la configuración del hardware debe editarse el archivo de canales `zapata.conf`, el cual especifica la configuración del dispositivo instalado y controla varias características y funcionalidades asociadas a los canales físicos, tales como: identificador de llamadas, llamada en espera, cancelación de eco y otras. A través de este archivo, se crean los canales lógicos TDM que serán utilizados en el plan de marcado. A continuación se muestra una parte del archivo `zapata.conf`:

```
Signalling= fxo_ks
Context= teléfonos
Channel=>1
Channel=>2
Signalling= Fxs_ks
Context= PSTN-entrante
Channel=>4
```

Como se aprecia en el segmento del archivo de configuración anterior, al canal 1 le corresponde la señalización de los módulos FXS, definida por el parámetro “signalling”. Esta opción se configuró utilizando el protocolo Kool start y señalización FXO. La configuración del módulo FXO se realizó con características similares.

3.6 Configurando un plan de marcado flexible y poderoso

En el archivo de configuración `extensions.conf` se define el plan de marcado de Asterisk, que permite realizar y recibir llamadas provenientes de la PSTN, definir teléfonos internos —ya sean de VoIP o teléfonos analógicos conectados a los puertos FXS—, especificar las extensiones y para conectar entre sí cada uno de los sistemas especificados. El plan de marcado de Asterisk se utiliza para lograr la interconexión entre los distintos protocolos a través de las aplicaciones. Se debe definir para cada canal VoIP un contexto particular y, a través de la inclusión de contextos, puede crearse un plan de marcado en donde todos los usuarios puedan comunicarse entre sí, sin importar el estándar que implementen. Un ejem-

plo de cómo quedaría el contexto de un estándar específico se presenta de la siguiente manera:

```
[SIP]
```

```
exten => _100x,1,Answer()
```

```
exten => _100x,n,Dial(SIP/
${EXTEN},10)
exten => _100x,n,Hangup
```

```
include => default
```

```
include=> PSTN-saliente
```

El comando “`exten =>`” es seguido por un número de extensión, una coma, la prioridad, otra coma y finalmente la aplicación. La mayor parte de las funcionalidades de Asterisk son creadas en forma de aplicaciones tales como: el buzón de voz, conferencia y el marcado, entre otras.

Las aplicaciones son parte fundamental en el sistema, ellas tratan al canal de voz, reproduciendo archivos de música, aceptando dígitos o terminando una llamada. Las aplicaciones disponibles en Asterisk se pueden ver usando el comando “`core show applications`” en el CLI.

Para lograr la comunicación entre todos los estándares, recibir y realizar llamadas hacia la PSTN se propone la creación de los siguientes contextos:

- ♦ PSTN-entrante

Este contexto se encargará de encaminar las llamadas provenientes de la PSTN, según sea necesario.

Debido a que el puerto FXO sólo puede ser comunicado con un sólo número, y una red VoIP puede tener una infinidad de estos, es necesario efectuar un IVR —*Interactive Voice Response / Respuesta de Voz Interactiva*— que pregunte al llamante a qué extensión quiere comunicar. De este modo, logra que se comunique con el usuario deseado. La configuración del contexto PSTN-entrante se describe a continuación:

```
[PSTN-entrante]
exten=>s,1,Answer()
exten =>s,2,Background(vm-
enter-num)
include => default
```

Como se puede apreciar, en el segmento anterior se utilizó la extensión especial “s” debido a que, cuando el módulo FXO detecta una llamada entrante, es imposible saber hacia qué extensión se dirige; por lo tanto, debe iniciar este contexto automáticamente.

♦ PSTN-saliente

Este contexto se encargará de las llamadas salientes de Asterisk, en este ejemplo se utiliza el canal 4 (Zap/4) de la tarjeta analógica.

En caso de que el puerto FXO esté conectado directamente a una PBX, es necesario lograr que Asterisk no confunda un número externo con uno interno, al igual que en las PBX tradicionales. En consecuencia, es conveniente definir un número de salida hacia ruta externa que usualmente es el número 9; por lo que se definió la extensión estándar “_9.”, que indica que todo número marcado que empiece con un 9 y después tenga uno o más caracteres, será dirigido al contexto [PSTN-saliente] como se muestra en el siguiente segmento:

```
[PSTN-saliente]
    exten => _9.,n,Dial(Zap/4/
    ${EXTEN},10)
    exten => _9.,n,Hangup
    include => default
```

3.6.1 Servicios avanzados del plan de marcado

Con los recursos avanzados del plan de marcado de Asterisk, se pueden proveer servicios de valor agregado a los usuarios, formando un sistema sencillo y poderoso que permite: transferencia de llamadas, conferencia, interfaz web para el administrador del sistema, grabación de llamadas, buzón de voz, entre otras facilidades. Para esto se requiere una configuración más compleja, donde deben editarse los archivos `manager.conf`, `http.conf`, `features.conf`, `meetme.conf`, `voicemail.conf`, entre otros.

Para que los mensajes de voz puedan ser archivados en cada buzón de voz de los usuarios, es necesario el uso de una base de datos que pueda manejar esta información. Asterisk utiliza un módulo intermediario entre la comunicación de este con la base de datos, llamado ODBC —*Open Database Connectivity*—. El conector ODBC es una capa de abstracción que posibilita la comunicación de Asterisk con una amplia gama de base de datos, por ejemplo, MySQL, PostgreSQL, etc. Las conexiones a las base de datos se configuran en el archivo `res_odbc.conf`, este archivo establece los parámetros que los módulos utilizarán para conectarse a estas.

3.7 Puesta en marcha del sistema

Para iniciar Asterisk basta con poner el código “Asterisk -vvvc” accediendo al CLI de Asterisk, en la cual se pueden ejecutar algunos comandos para la gestión, administración y control del sistema. Mediante el comando “Asterisk -h” se puede explorar el rango completo de opciones. En la tabla II se muestran los parámetros más utilizados.

Parámetros	Descripción
-h	Muestra las opciones de códigos disponibles
-c	Inicia Asterisk con una interfaz de línea de comando (*>CLI)
-v	Verbosidad, incluye mensajes detallados
-r	Se conecta a una consola de Asterisk ya iniciada
-f	Inicia Asterisk pero no lo pone en modo consola

Tabla 2 Parámetros más utilizados de las opciones de Asterisk [6].

3.7.1 Escenarios de uso

Es bastante común en estos días el uso de *softswitch* o PBX, pero su comercialización se hace de forma independiente para cada función del sistema. Sus componentes se realizan por separado y, muchas veces, tienen diferentes fabricantes. Los costos de adquisición de cada uno de estos componentes son elevados y la integración, en reiteradas ocasiones, es difícil. Asterisk realiza estas funciones de modo integrado, el licenciamiento es gratuito y puede ser hecho en un único o en varios servidores de acuerdo con un dimensionamiento apropiado.

En la figura 2 se muestra el escenario más común para pequeñas o medianas empresas. Este tipo de escenario admite la interconexión entre las distintas normas de señalización incluyendo la PSTN. Además, se pudiera conectar Asterisk a una PBX en vez de hacerlo directamente hacia la Red de Telefónica Pública Conmutada.

Asterisk puede convertir las señales analógicas (FXS, FXO) o digitales (RDSI) que vienen de la central telefónica, o de los teléfonos de clientes en voz sobre IP y transmitirlos por la red de datos. La convergencia propicia la reducción del número de circuitos y un mejor aprovechamiento de los recursos. Una pasarela permitirá que las conexiones en telefonía analógica puedan ser convertidas en voz sobre IP y transmitidas por la red de datos a otra oficina sin pasar por la tarificación de la red pública.

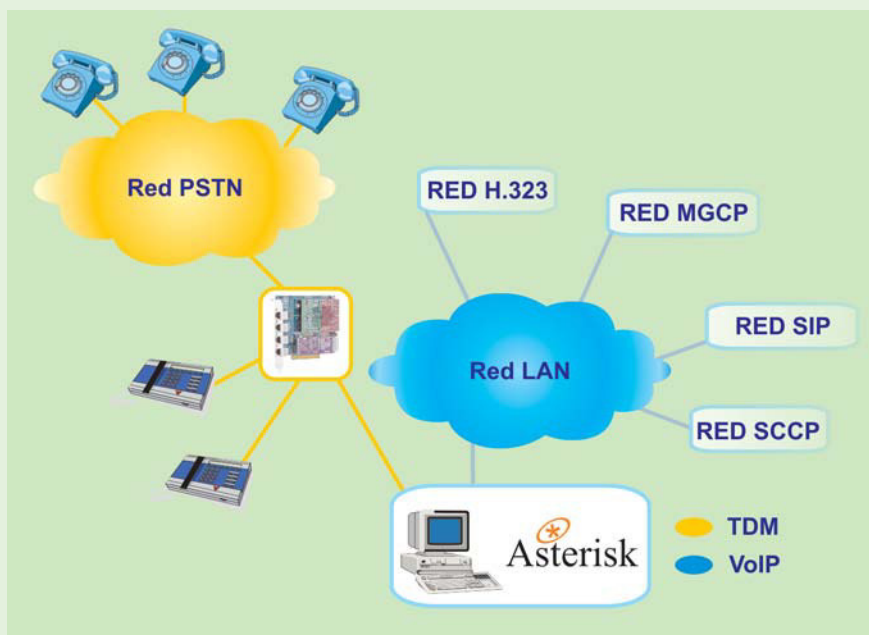


Figura 2 Escenarios de uso de Asterisk. (Fuente: elaboración propia).

3.8 Requerimientos técnicos para la ubicación del servidor Asterisk

La ubicación del servidor dependerá de los factores externos que influyen en su funcionamiento, tales como la humedad, el polvo, interferencia, temperatura y, muy importante, las consideraciones de seguridad. Por consiguiente, se recomienda alejar el servidor de áreas con humedad, limpiar el local regularmente y ubicarlo en un lugar con una temperatura adecuada.

La seguridad del servidor debe realizarse contra intrusiones en la red utilizando detectores de intrusos locales o de red tales como el SNORT o el OSSEC. Además de tener en consideración la seguridad física del equipo para protegerlo del acceso del personal no autorizado en áreas restringidas.

Uno de los grandes retos al implementar VoIP, especialmente en regiones en desarrollo, es garantizar que exista un ancho de banda constante para las conversaciones. Con el propósito de ofrecer una buena calidad en la conversación, el ancho de banda que necesitan los dos flujos de tráfico se debe garantizar con independencia del estado del resto de las conexiones. Cuando se diseña una red de VoIP, ha de intentarse la optimización del ancho de banda, el control de las fluctuaciones de la red (*jitter*) y la disminución de la latencia.

Teniendo en cuenta los elementos expuestos, es posible crear, con un conocimiento básico de Asterisk, sistemas de telefonía así como PBX IP o pasarelas basadas en este poderoso software. Además de establecer medios capaces de lograr la interoperabilidad entre los distintos estándares de señalización de llamada existentes de forma transparente y eficiente. También, deben ser capaces de crear conexiones con líneas de acceso analógicas hacia la PSTN, a través de las tarjetas analógicas de la compañía Digium.

4 Resultados y discusión

Para evaluar el cumplimiento del objetivo principal de este trabajo, se realizó el diseño de un escenario de prueba en el cual se experimentó con un conjunto de clientes evaluando *codecs*, alternativas de configuración y funciones. En cuanto a los protocolos de señalización, las pruebas se realizaron

con el empleo de los protocolos SIP, IAX2, H323 y SCCP. Las pruebas abarcaron la evaluación de la instalación y configuración, las características del sistema, la facilidad en el uso de las interfaces de usuario y los protocolos de señalización soportados por el sistema para la función de pasarela. Durante las llamadas realizadas se chequearon la calidad percibida, la capacidad de conexión y cualquier problema en el manejo de las mismas.

Con el propósito de comprobar la interoperabilidad entre los protocolos se realizaron llamadas entre los *softphones* X-Lite (SIP), ZoIPer (IAX2), SJphone (H.323) y VTGO-PC (SCCP). Además se probaron los servicios de conferencia, menú IVR, buzón de voz, música en espera, transferencia y grabación de llamadas. Se demostró, así, el buen funcionamiento de Asterisk como una PBX IP de servicios avanzados. También, se comprobaron los diferentes contextos y extensiones especiales que indicaron ser un sistema con una gran variedad de posibilidades y configuraciones.

En el departamento de telecomunicaciones y telemática del Instituto Superior Politécnico José Antonio Echeverría, se realizaron un conjunto de pruebas para comprobar la funcionalidad de Asterisk como pasarela entre la PSTN y una red VoIP. Se utilizaron 5 PC, una como servidor, el cual llevará instalado en su bus PCI una tarjeta del tipo TDM22B y, en las otras, los diferentes *softphones* sobre Windows XP. El servidor corrió bajo el SO Debian en una Intel Celeron a 1.2 GHz con 512 MB de RAM, tarjeta de red a 100 Mbit/s y conexión a la LAN de la CUJAE.

Un puerto FXO de la tarjeta se conectó directamente a la PSTN y, en los puertos FXS, se conectaron 2 teléfonos analógicos LG Electronics Inc. modelo GS-5140 de ETECSA como se puede apreciar en la figura 2.

Como resultado se obtuvo que la configuración de Asterisk, como Pasarela de Entronques, es bastante adaptable y poderosa; no obstante, es necesario tener un cuidado extremo con la manipulación de los módulos y la tarjeta. En las pruebas realizadas se alcanzó una calidad de voz aceptable, aunque en la comunicación entre los teléfonos analógicos conectados a los puertos FXS se escuchó un poco de eco.

5 Conclusiones

En este trabajo se ha propuesto una metodología para el diseño de pasarelas de VoIP con acceso a la PSTN, además se analizaron los conceptos básicos de la telefonía IP, haciendo énfasis en las pasarelas de medios como elementos imprescindibles en la migración hacia las Redes de Nueva Generación. Mediante esta investigación se presentaron soluciones y herramientas para el desarrollo de PBX IP y pasarelas basadas en sistemas abiertos que pueden reducir sus elevados costos y mejorar la capacidad de introducción de nuevas prestaciones.

En cuanto a todas las pruebas realizadas se pudo concluir que, Asterisk es un software de configuración compleja; sin embargo, es de fácil manejo gracias a la diversidad de documentación y aplicaciones que aparecen en Internet. Es una aplicación de excelente calidad que puede ejecutarse en servidores de bajas prestaciones, y resulta una alternativa real para soluciones de alto nivel en diferentes universidades, empresas nacionales y las FAR —Fuerzas Armadas Revolucionarias— que requieran una solución óptima de telecomunicaciones. Esto logrará elevar los niveles de seguridad y estabilidad de los sistemas al permitir que el país tienda, indiscutiblemente, hacia la necesaria independencia tecnológica. 

6 Referencias bibliográficas

- [1] Lenahan, F. "Una visión práctica de la evolución de las redes del futuro". Telcordia Technologies, 2005. <http://creativecommons.org/licenses/byncsa/2.5/>. (acceso marzo 8, 2008).
- [2] Lenahan, F. "Una visión del futuro: impactos radicales en el negocio de comunicaciones". Telcordia Technologies, 2005. <http://creativecommons.org/licenses/byncsa/2.5/>. (acceso marzo 8, 2008).
- [3] ITU-T T-REC H.248-1 Protocolo de control de las pasarelas: Versión 3," ITU-T, 2006.
- [4] Escudero, A. y Berthilson, L. VoIP para el desarrollo: una guía para crear una infraestructura de voz en regiones en desarrollo. 2006. http://www.wilac.net/doc/tricalcar/materiales_abril2008/PDF_es/I6_es_voip_presentacion_v02-1.pdf. (acceso julio 8, 2008).
- [5] Van Meggelen, J. y Madsen, L. *Asterisk: The Future of Telephony*. USA: O'Reilly, 2007, págs. 404.
- [6] Goncalvez, F. *Asterisk PBX Guía de la configuración*. Río de Janeiro, Brasil: V. Office Networks, 2007, págs. 362.
- [7] Capote, J. A. y Capote, A. M., "Implementación de una pasarela multiprotocolo utilizando la plataforma telefónica código abierto Asterisk™". Tesis de Diploma, Departamento de Telecomunicaciones, Facultad de Eléctrica, Instituto Superior Politécnico José Antonio Echeverría, Ciudad de La Habana, Cuba, 2008.