

Comunicaciones Unificadas con Elastix

Volumen 1

Edgar Landívar

Copyright (c) 2008-2009 Edgar Landívar

Este documento está permitido de copiar, distribuir y/o modificar bajo los términos de la licencia **GNU Free Documentation License, Versión 1.3**; sin Secciones Invariantes (*Invariant Sections*), Textos de Cubierta Frontal (*Front-Cover Texts*), y sin Textos de Cubierta Posterior (*Back-Cover Texts*).

El autor puede en cualquier momento decidir actualizar la licencia de la presente obra a una versión más reciente de la **GNU Free Documentation License**.

Una copia de esta licencia se incluye en el Apéndice A: *GNU Free Documentation License*.

Segunda Edición, versión Beta

*A los que siempre creyeron y a otros gigantes que nos
mostraron el camino*

Reconocimiento

La elaboración de este libro involucró un enorme trabajo que fue facilitado gracias a la generosa ayuda de algunas personas con las cuales me encuentro agradecido. Mis más sinceros agradecimientos a:

- *A la comunidad Elastix*, por su valioso *feedback* del producto Elastix y de seguro por el que vendrá acerca de esta obra. Sin este componente este libro no podría evolucionar
- *Edwin Boza*, por su ayuda en el capítulo de instalación de Elastix sobre *soft RAID*
- *Bruno Macías*, por aportar con contenido para el apartado de *Endpoint Configurator*
- *Jorge Gutierrez*, por colaborar con contenido para la configuración de teléfono IP Polycom
- *Rafael Bonifaz*, por disipar mis dudas acerca de la FDL

Por qué comprar este libro?

Comprar una versión impresa de este libro es una manera de colaborar con el proyecto Elastix. Sin embargo, si al lector no le gustan los libros impresos pero desea colaborar con este proyecto de software libre, todavía puede realizar una donación a través del siguiente enlace:

http://www.elastix.org/index.php?option=com_content&task=view&id=128&Itemid=54

Qué significa “versión Beta”?

Significa que se trata de una edición preliminar que pudiera estar sujeta a errores menores y por lo tanto cualquier retroalimentación de los lectores es bienvenida.

Feedback

Cualquier sugerencia o colaboración será bien recibida. Que el lector no dude en escribir un email a elastixbook@palosanto.com

Indice

| | |
|---|-----------|
| Indice | 7 |
| Introducción..... | 13 |
| 1. Introducción a la telefonía | 15 |
| 1.1 Breve historia de la telefonía | 15 |
| Los albores de la telefonía | 15 |
| Los dolores de cabeza de las famosas patentes | 16 |
| El desarrollo de la tecnología telefónica..... | 18 |
| 1.2 Principios y transmisión de la voz humana | 18 |
| Rango de frecuencias de la voz humana..... | 19 |
| El micrófono | 19 |
| Ancho de banda y capacidad de información..... | 21 |
| 1.3 Digitalización de la voz | 21 |
| Teorema de Nyquist..... | 22 |
| 1.4 Redes orientadas a circuitos | 22 |
| 1.5 Redes orientadas a paquetes | 23 |
| 1.6 Red Pública Telefónica (<i>PSTN</i>)..... | 24 |
| 1.7 Circuitos analógicos | 24 |
| Señalización analógica | 25 |
| DTMFs | 27 |
| El teléfono analógico | 28 |
| 1.8 Circuitos digitales | 30 |
| La base DS-0 | 30 |
| Circuitos T-carrier y E-carrier | 30 |
| SONET y Circuitos Opticos | 30 |
| 1.9 Protocolos de Señalización Digital..... | 30 |
| Señalización Asociada al Canal (CAS) | 31 |
| Señalización de Canal Común (CCS)..... | 31 |
| 2. Introducción a la VoIP | 33 |
| 2.1 VoIP: una sopa de protocolos..... | 34 |
| Clasificando los protocolos VoIP | 34 |
| 2.2 Protocolo IP | 36 |
| El Internet | 36 |
| Qué es el protocolo IP?..... | 37 |
| Dirección IP | 37 |
| Paquete IP | 37 |
| Direccionamiento IP | 38 |

| | |
|---|-----------|
| 2.3 Protocolos de transporte | 39 |
| Protocolo TCP | 40 |
| Protocolo UDP..... | 40 |
| 2.4 Codificación de la voz..... | 41 |
| Codecs | 41 |
| 2.5 Sobrecarga de protocolos..... | 43 |
| Comparativa de <i>codecs</i> | 44 |
| 3. Linux para Administradores de Elastix | 47 |
| 3.1 Introducción..... | 47 |
| 3.2 Línea de comandos de Linux..... | 48 |
| 3.3 Comandos básicos | 49 |
| 3.4 Sistema de archivos | 51 |
| Organización..... | 51 |
| Permisos | 52 |
| 3.5 RPMs y actualizaciones vía Yum..... | 56 |
| Yum al rescate | 57 |
| 4. Asterisk Essentials | 59 |
| 4.1 Qué es Asterisk? | 59 |
| 4.2 Breve historia de Asterisk | 60 |
| 4.3 Funcionalidad provista por Asterisk..... | 60 |
| 4.4 Funcionamiento de Asterisk | 66 |
| Encendido y apagado..... | 66 |
| Directorios de Asterisk | 67 |
| Estructura modular | 67 |
| 4.5 El proyecto Zaptel, ahora DAHDI..... | 68 |
| Breve historia del proyecto Zaptel..... | 69 |
| 4.6 Configuración de Asterisk..... | 70 |
| Archivos de configuración | 70 |
| Comentarios en los archivos de configuración..... | 70 |
| 4.7 Plan de marcado (dial plan) | 71 |
| Contextos..... | 71 |
| Instrucciones..... | 72 |
| Variables..... | 74 |
| Aplicaciones más comunes..... | 76 |
| 4.8 Asterisk CLI | 77 |
| Comandos del CLI..... | 79 |
| 5. Instalando Elastix | 81 |
| 5.1 Instalando desde CD | 81 |
| 5.2 Instalando Elastix sobre un CentOS o Red Hat preinstalado | 87 |

| | |
|--|------------|
| 5.3 Instalando sobre un <i>soft</i> RAID | 87 |
| 6. Elastix Overview | 93 |
| 6.1 Breve historia del proyecto..... | 94 |
| 6.2 Características de Elastix..... | 95 |
| VoIP PBX..... | 95 |
| Fax | 96 |
| General | 96 |
| Email..... | 97 |
| Colaboración..... | 97 |
| Extras..... | 97 |
| Call Center..... | 97 |
| Mensajería instantánea | 97 |
| 6.3 Licenciamiento | 98 |
| 6.4 Sobrevuelo por la interfase de administración Web de Elastix | 98 |
| System | 98 |
| PBX | 100 |
| Fax | 100 |
| Email..... | 101 |
| IM | 101 |
| Reports..... | 101 |
| Extras..... | 102 |
| Agenda..... | 102 |
| 6.5 Elastix.org: Sitio oficial del proyecto | 103 |
| Foros | 103 |
| 7. Elastix: Configuración Básica | 105 |
| 7.1 Ingreso al Sistema..... | 105 |
| 7.2 Configuración de red | 105 |
| 7.3 Creando una extensión | 106 |
| Configuración de teléfono <i>softphone</i> | 109 |
| 7.4 Configuraciones generales..... | 109 |
| Opciones del comando Dial (<i>Dial command options</i>)..... | 111 |
| 7.5 Configuración de colas | 111 |
| 7.6 IVR y sistema de grabación..... | 115 |
| Grabación de mensaje de bienvenida | 116 |
| Configurar un IVR de bienvenida | 117 |
| 7.7 Salas de conferencias..... | 119 |
| 7.8 Parqueo de llamadas | 121 |
| 7.9 Música en espera | 123 |
| 7.10 Flash Operator Panel (FOP) | 124 |

| | |
|--|------------|
| 7.11 Funciones de transferencia de llamadas | 125 |
| Transferencia ciega..... | 125 |
| Transferencia atendida..... | 125 |
| 8. Configuración de hardware..... | 127 |
| 8.1 Tipos de tarjetería y fabricantes..... | 127 |
| 8.2 Instalando físicamente la tarjetería | 128 |
| Hardware modular | 129 |
| 8.3 Detección de hardware | 129 |
| 8.4 Configuración de interfaces digitales a bajo nivel..... | 130 |
| Configuración de los <i>drivers</i> de Zaptel | 131 |
| 8.5 Configuración de troncales..... | 132 |
| zapata.conf..... | 133 |
| 8.6 Algunos cambios que se darán con el cambio de Zaptel a DAHDI | 136 |
| Algunos cambios de nombre a tener en cuenta | 137 |
| 9. Configuración de teléfono IP..... | 139 |
| 9.1 Configuración de <i>softphone</i> | 140 |
| 9.2 Configuración de teléfono físico | 143 |
| Configuración de red | 144 |
| Registro del dispositivo con el servidor Elastix | 146 |
| 9.3 Utilizando el <i>endpoint configurator</i> | 150 |
| Configuración de un teléfono ATCOM, modelo AT-530 | 152 |
| Configuración de un teléfono Polycom SoundPoint IP 501 | 154 |
| Actualización por TFTP | 156 |
| Personalizando los archivos de configuración | 156 |
| 10. Elastix: Configuración avanzada | 159 |
| 10.1 Ruteo de llamadas..... | 159 |
| Rutas Entrantes | 159 |
| Rutas Salientes..... | 162 |
| 10.2 Troncales | 163 |
| Troncales de tipo IAX y SIP..... | 165 |
| Troncales de tipo ZAP..... | 166 |
| Troncales personalizadas..... | 166 |
| 10.3 Tarifación | 166 |
| Visualizar, Editar y Eliminar | 167 |
| Crear Tarifa | 167 |
| Reporte de Facturación..... | 168 |
| Distribución de Destinos | 169 |
| Configurar Facturación..... | 171 |
| 10.4 Reportes..... | 172 |

| | |
|---|------------|
| Reporte CDR | 172 |
| Uso de Canales | 173 |
| 10.5 Configuración DISA..... | 174 |
| 10.6 Condiciones de tiempo | 175 |
| 11. Funcionamiento interno de freePBX | 179 |
| 11.1 Fundamentos de freePBX..... | 179 |
| 11.2 Organización de archivos | 180 |
| 11.3 Configuración | 181 |
| amportal.conf..... | 181 |
| 11.4 retrieve_conf: generador del plan de marcado | 182 |
| 11.5 dialparties.agi..... | 182 |
| 11.6 Estructura de un módulo de freePBX..... | 183 |
| 12. Interconexión con Elastix remotos..... | 185 |
| 12.1 A través de troncales SIP..... | 185 |
| Definiendo la troncal | 186 |
| Definiendo la ruta saliente..... | 187 |
| 12.2 Conexión con proveedores de llamadas vía IP | 187 |
| Definiendo la troncal | 188 |
| Definiendo la ruta saliente..... | 188 |
| 13. Calidad de voz..... | 191 |
| 13.1 Síntomas que afectan la calidad de voz..... | 191 |
| Eco..... | 191 |
| Bajo nivel o volumen..... | 193 |
| Retardo | 193 |
| Distorsión de la voz | 193 |
| Comunicación entrecortada | 194 |
| 13.2 Parámetros relacionados con la calidad de voz en redes de paquetes | 194 |
| Retardo de red..... | 194 |
| Pérdidas de paquetes..... | 195 |
| Jitter | 195 |
| 13.3 Cómo medir la calidad de voz | 197 |
| Escala MOS | 198 |
| Modelo E | 198 |
| 13.4 Canceladores de eco | 202 |
| Cómo funciona un cancelador de eco? | 202 |
| El estándar G.168 | 203 |
| 14. Protocolos VoIP al descubierto | 205 |
| 14.1 Protocolo SIP..... | 205 |
| Detalles del protocolo..... | 206 |

| | |
|---|------------|
| Señalización SIP: Métodos y respuestas | 206 |
| SIP y Asterisk | 207 |
| 14.2 Protocolo IAX | 207 |
| Fases de una llamada IAX | 208 |
| Frames IAX | 209 |
| 14.2 Protocolo RTP | 210 |
| Estructura de un paquete RTP | 211 |
| 15. Diagnóstico básico de problemas | 213 |
| 15.1 Interpretando el <i>log</i> | 213 |
| El archivo /var/log/asterisk/full | 215 |
| Qué información nos puede indicar el campo TEXTO? | 216 |
| Número de proceso | 217 |
| 15.2 Diagnóstico a través del CLI | 217 |
| 15.3 <i>Sniffing</i> | 217 |
| Captura de llamadas..... | 218 |
| 16. Control externo de Asterisk..... | 219 |
| 16.1 Asterisk AGI..... | 219 |
| Ejecución de <i>scripts</i> AGI..... | 220 |
| Parámetros enviados al <i>script</i> AGI..... | 220 |
| Comandos AGI..... | 221 |
| Librerías AGI..... | 223 |
| Ejemplo sencillo de <i>script</i> AGI | 223 |
| 16.2 Asterisk AMI | 224 |
| Probando desde <i>telnet</i> | 225 |
| Tipo de paquetes AMI | 226 |
| Comandos o acciones AMI | 227 |
| Apéndice A. GNU Free Documentation License | 229 |
| Apéndice B. Archivos de configuración de Asterisk distribuidos con Elastix | 239 |
| Apéndice C. Lista de comandos de CLI | 249 |

Introducción

Ya existían otros libros o manuales de Elastix cuando me vi en la misión de Escribir la presente obra. Había leído ya *Elastix Without Tears*, un gran trabajo de Ben Shariff y me parecía que ya estaban cubiertos los temas más importantes cuando nos reunimos en mi oficina con otros colaboradores a planificar lo que sería nuestro nuevo producto de entrenamiento para la certificación de Elastix.

Casi de inmediato nos dimos cuenta que pese a que las obras existentes cubrían muchos tópicos importantes no funcionarían del todo como material de estudio para un curso de certificación. La obra que necesitábamos tenía que cubrir aspectos internos del funcionamiento de Elastix y sus componentes, así como también cubrir posibles vacíos de los alumnos en tópicos diversos como Asterisk y protocolos de comunicación como SIP y RTP. No existía nada parecido en ese momento, una alternativa era usar diferentes obras para cubrir la totalidad del entrenamiento pero eso sería un poco engorroso y se vería algo desorganizado.

Salimos de la reunión con el sinsabor de no tener material de estudio.

Semanas después me di a la tarea de escribir un pénsum para nuestros cursos olvidándome un poco del tema del libro faltante. El pénsum fue del agrado de todos así que decidí bosquejar brevemente lo que debería contener cada tópico y ese fue el inicio del libro que tienen en sus manos (o en su pantalla)... Sin darme cuenta lo había comenzado a escribir. Una noche en mi casa vi que ya tenía cierta forma así que decidí comentarlo al siguiente día en la oficina y todos aprobaron la idea. Decidimos hacer dos tomos o volúmenes, el primero cubriría los tópicos requeridos para nuestro primer curso y un segundo cubriría los tópicos de nuestro segundo y final curso.

Ya iba por la mitad del primer volumen cuando surgió la pregunta acerca del licenciamiento de la obra. Elastix es software libre y yo no había pensado aún en el licenciamiento del libro. Algunos pensaban en que no debería ser libre debido a que era material de entrenamiento que de alguna manera nos otorgaba ventaja competitiva en el mercado. En este punto debo de agradecer la ayuda de algunos integrantes del equipo de PaloSanto Solutions por sus acertados argumentos en pro de FDL (*Free Documentation License*).

Finalmente la presente obra se termino con la escritura de esta introducción el 29 de Octubre de 2008. Espero que sea de la utilidad y agrado de todos vosotros.

-- Edgar Landivar
Elastix Project Leader

1

Introducción a la telefonía

No importa si yo muero, otros terminarán mi trabajo...

-- Mark Twain

1.1 Breve historia de la telefonía

Sin duda la invención de lo que hoy conocemos como telefonía debió ser un acto asombroso en su tiempo, casi mágico. El oír la voz de alguien remoto en tiempo real saliendo de una misteriosa caja en una época en la que esto era solo posible en la ciencia ficción debió haber sido una experiencia única y casi fantástica.

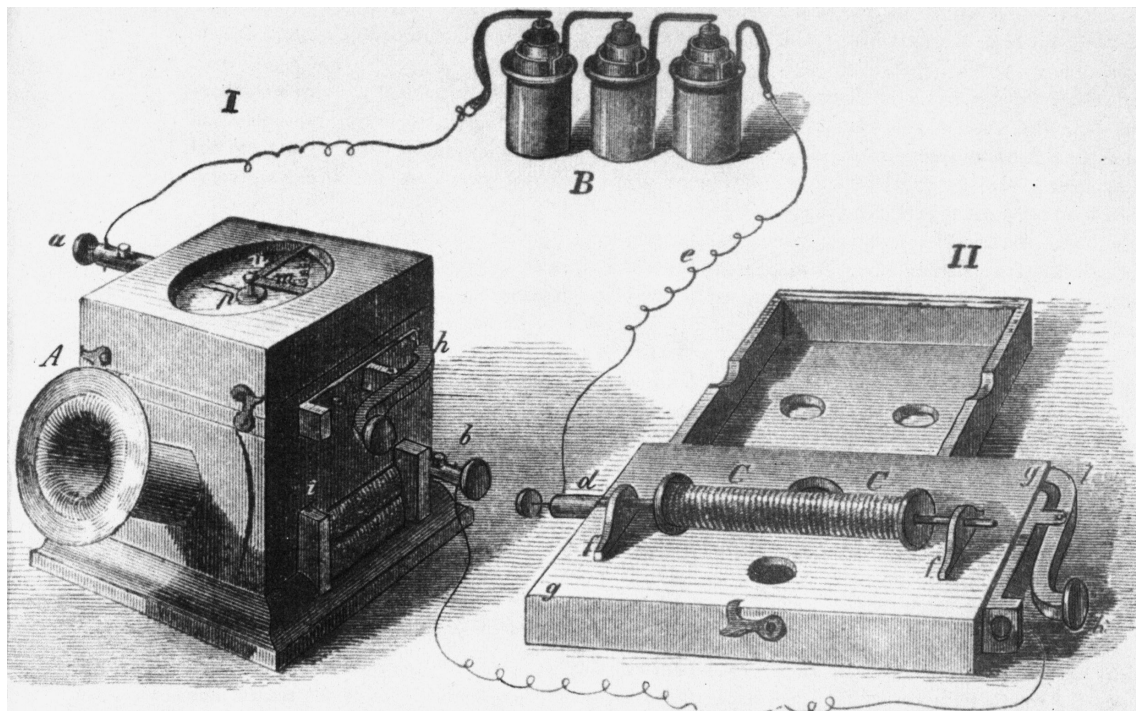
De seguro que la idea de un aparato parlante ya rondaba por la cabeza de muchos inventores desde tiempos inmemoriales, pero fue a inicios del siglo 19 cuando parecía algo alcanzable pues ya se había descubierto la electricidad, inventado el telégrafo y algunos ya experimentaban con ondas de radio.

Los albores de la telefonía

A mediados del siglo 19 hubo un interesante movimiento en torno a lo que hoy conocemos como teléfono.

En 1849 Antonio Meucci, médico italiano considerado por muchos como el inventor del teléfono, hizo una demostración de un dispositivo capaz de transmitir voz en La Habana. Pocos años después, en 1854, el mismo Meucci hace una nueva demostración de su invención en la ciudad de Nueva York.

Mientras Meucci se las daba a las tareas de inventor, otros también perseguían la idea de construir un “telégrafo parlante” y es así como en 1860 el alemán Johann Philipp Reis construye un dispositivo capaz de transmitir voz basado en la idea original de Charles Bourseul, quien a su vez describió la construcción de dicho dispositivo en 1854 pero nunca lo construyó. Reis continuó mejorando su aparato y un año más tarde ya estaba transmitiendo voz a más de 100 metros de distancia.



Teléfono de Reis

Un par de años más tarde Innocenzo Manzetti construye el esperado “telégrafo parlante” que él mismo había visionado ya en 1844, pero no se interesa en patentarlo.

Los dolores de cabeza de las famosas patentes

Hasta aquí ya existían algunos prototipos de teléfono pero nadie lo había patentado.

El primero en tratar de patentar el invento fue Meucci, quien en 1871 suscribió un documento de “aviso de patente” pero por su condición económica nunca pudo pagar el dinero para terminar este trámite y su “aviso de patente” expiró pocos años después.

A Meucci no le fue muy bien que digamos pues no pudo vender su invento y alcanzar la prosperidad.

En 1875, un año después de expirar el trámite de patente de Meucci, Alexander Graham Bell, un escocés radicado en los Estados Unidos, logra patentar un aparato similar y es el primero en hacerlo.

Bell había estado experimentando previamente con algunas ideas para concebir su dispositivo telefónico hasta que un día logró arrancarle a la electricidad algunos sonidos. Cuenta la historia que la primera llamada que hizo fue para decirle a su asistente las célebres frases “Sr. Watson, venga, necesito verlo” (*Mr. Watson... come here... I want to see you*).

Un hecho curioso que desató mucha polémica es que otro inventor llamado Elisha Gray también trató de patentar un invento similar tan solo unas pocas horas después de Bell. Los dos inventores entraron en una conocida disputa legal que finalmente Bell ganó.

Gracias a la patente Bell pudo hacer de la idea del teléfono un negocio rentable y tiene el mérito de haber desarrollado la idea y convertirla en algo práctico para la sociedad.

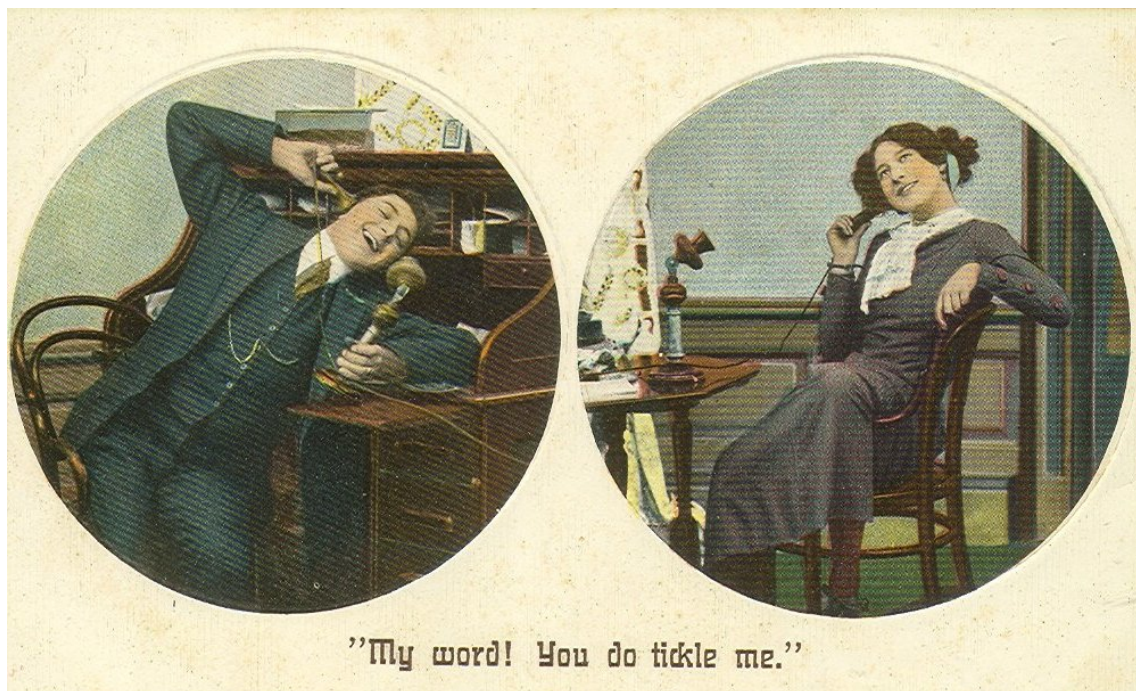


Ilustración de dos personas hablando por teléfono a fines del siglo XIX

Se cuenta que en determinado momento Bell trató de vender su patente a *Western Union* por \$100 mil dólares pero el presidente de Western Union se negó pues consideró que el teléfono “era nada más que un juguete”. Tan solo dos años más tarde el mismo directivo de Western Union le comentó a sus colegas que si pudiera conseguir la patente de Bell por \$25 millones de dólares lo consideraría una ganga.

Esto nos da una idea de cómo comenzaba a crecer el negocio de Bell. En 1886, ya existían más de 150,000 abonados telefónicos en los Estados Unidos.

A partir de aquí la telefonía poco a poco se empezó a convertir en un servicio básico de la sociedad actual.

El desarrollo de la tecnología telefónica

Como sucede siempre con los avances tecnológicos la telefonía continuó evolucionando. Al principio, para que un abonado se comunicara con otro este tenía que solicitarle la llamada a una operadora, quien manualmente conectaba los cables para conmutar un punto con otro. En 1891 se inventó un teléfono “automático” que permitía marcar directamente.

En un principio Bell fue casi exclusivamente la única compañía en explotar la tecnología debido a sus patentes. Sin embargo, cuando estas expiraron nacieron cientos de pequeñas compañías que empezaron a dar servicio, la mayoría en sitios rurales donde Bell aún no llegaba. Poco a poco estas compañías empezaron a crecer y ya a inicios del siglo 20 tenían en su conjunto más abonados que la propia Bell. La sana competencia hizo lo suyo y la tecnología telefónica aceleró su evolución.

Ya para finales de la segunda guerra mundial el servicio telefónico llegaba a millones de abonados.

En 1947, científicos de Bell inventan el transistor y cambian el curso de la historia de la humanidad. En 1948 ganan el Premio Nobel por su trabajo.

En los años 60s se lanzan los primeros satélites de comunicaciones y las comunicaciones entre continentes se facilitan. No está demás decir que esto no hubiera sido posible sin la previa invención del transistor.

1.2 Principios y transmisión de la voz humana

La voz humana está compuesta por ondas acústicas que viajan a través del aire a la velocidad del sonido, esto es a 1,244 Km/h (o 340 m/s). Bastante rápido verdad? Incluso más rápido que un avión comercial. Pero esta rapidez no significa que me pueda comunicar fácilmente con puntos distantes pues la voz humana se atenúa rápidamente, perdiendo energía a medida que viaja. Luego de unos pocos metros ya no podemos escuchar una conversación.

La voz humana por tanto es de la misma naturaleza que el resto de ondas acústicas y esto ya se conocía desde antes de la invención del teléfono.

Antes de la invención del teléfono también se conocía que existían otros tipos de ondas llamadas ondas eléctricas que podían ser transmitidas a través de un conductor metálico como un cable de cobre. Este segundo tipo de ondas es de una naturaleza diferente a las ondas acústicas y viaja a la velocidad de la luz, es decir aproximadamente 300,000 km/s. Es decir, más de lo que podemos imaginar; casi instantáneamente desde un punto de vista terrenal. Adicionalmente podemos controlar la atenuación de estas ondas y hacerlas viajar por grandes distancias.

Con estos hechos conocidos ya a mediados del siglo 19 es más fácil comprender que muchos persiguieran la idea de transformar las ondas acústicas en ondas eléctricas para así poder transmitir las luego a grandes distancias a través de conductores metálicos. La cuestión es que había que inventar un dispositivo para hacer dicha transformación y allí estaba la clave del asunto. Este dispositivo, conocido como micrófono en nuestros días es una parte importante de cualquier aparato telefónico.

Rango de frecuencias de la voz humana

Otra característica importante de la voz humana es que las cuerdas vocales modulan la voz en un amplio espectro de frecuencias que van de graves a agudos en un rango aproximado de 20Hz a 20kHz. Todo un abanico de sonidos!

Esto nos hace suponer que un micrófono debe ser capaz de capturar y transmitir todo este rango de frecuencias. Sin embargo, en la actualidad sabemos que para transmitir voz "entendible" no es necesario transmitir todas las frecuencias sino un rango mucho menor y transmitir un rango menor de frecuencias tiene sus ventajas pues facilita la transmisión como veremos más adelante. Por lo tanto los teléfonos comerciales solo transmiten un rango aproximado de 400Hz a 4kHz. Esto distorsiona un poquito la voz pero de todas maneras se puede entender. Es por eso que cuando oímos a alguien por teléfono su voz suena ligeramente diferente que en la vida real pero aun así podemos entender la conversación.

El micrófono

El micrófono fue un elemento clave en la invención del teléfono pues era el dispositivo que realizaba la conversión de las ondas mecánicas a ondas eléctricas.

Hay muchos tipos de micrófonos que operan sobre diferentes principios. Uno que se usó por mucho tiempo en teléfonos era el de carbón que consistía en una cápsula llena de granitos de carbón entre dos placas metálicas. Una de las placas era una membrana que vibraba con las ligeras presiones de las ondas de voz; de esta manera la resistencia

eléctrica de la cápsula variaba con la voz y se generaba una señal eléctrica correspondiente.



Típico micrófono de carbón extraído de teléfono de disco

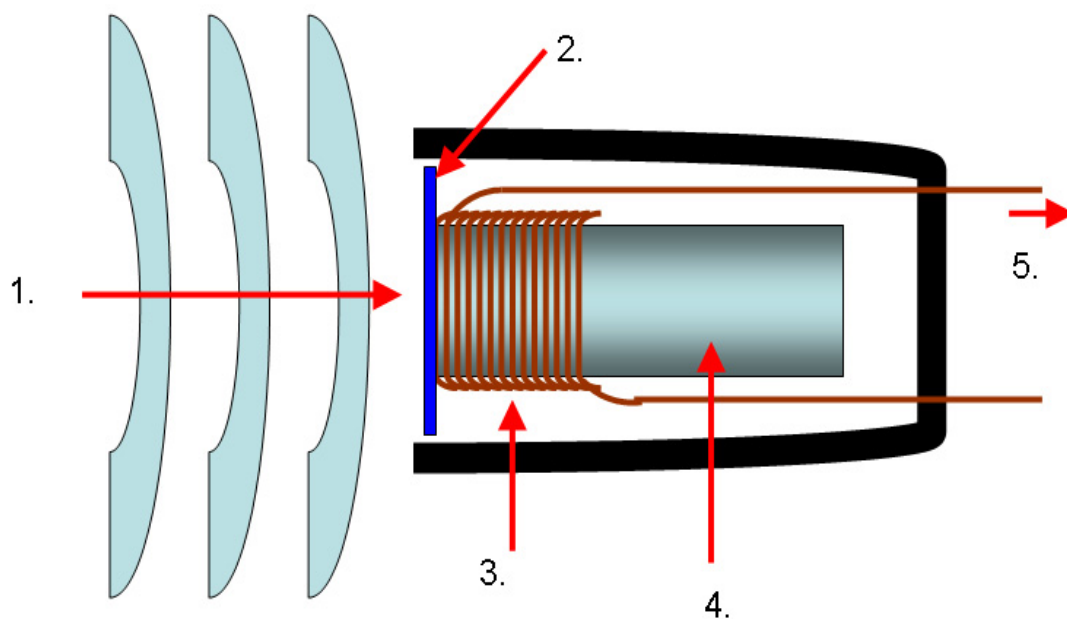


Diagrama esquemático de micrófono electro-magnético

Otro tipo de micrófono muy común en la actualidad es el dinámico o electro-magnético que consiste en una bobina de hilo de cobre enrollada sobre un núcleo de material ferromagnético. Este núcleo se encuentra sujeto a un diafragma que vibra con la presión de las ondas de voz. De esta manera se induce una ligera corriente eléctrica en la bobina que es amplificada luego al interior del teléfono.

En la figura anterior podemos observar algunos componentes del micrófono electro-magnético reaccionando frente al estímulo de las ondas de voz.

- 1 Ondas de voz
- 2 Diafragma
- 3 Bobina
- 4 Núcleo ferromagnético
- 5 Corriente inducida

Ancho de banda y capacidad de información

Ancho de banda es un término algo difícil de entender al principio pues es un concepto bastante amplio.

En general podemos decir que ancho de banda es una medida de la cantidad de información que podemos transmitir por un medio por unidad de tiempo. Debido a que es una medida por unidad de tiempo muchas veces se hace una analogía con la velocidad. Pero hay que estar atento a confusiones.

Medidas comunes para expresar el ancho de banda son los bits por segundo. Esta medida también equivale a bits/s, bps o baudios.

El ancho de banda es un término muy importante cuando se habla de telefonía pues las comunicaciones en tiempo real necesitan un ancho de banda mínimo asegurado para entregar una comunicación de calidad en destino.

1.3 Digitalización de la voz

Las redes digitales de transmisión de voz y datos son comunes en nuestra era. Fueron creadas ya que presentan ciertas ventajas sobre las redes analógicas como por ejemplo que conservan la señal casi inalterable a través de su recorrido. Es decir que es más difícil que la comunicación se vea afectada por factores externos como el ruido eléctrico. Además nos provee de métodos para verificar de cuándo en cuando la integridad de la señal, entre otras ventajas.

Dicho fácil, digitalizar una señal de voz no es otra cosa que tomar muestras (a intervalos de tiempo regulares) de la amplitud de la señal analógica y transformar esta información a binario. Este proceso se denomina muestreo.

Teorema de Nyquist

En 1928 Henry Nyquist, un ingeniero Suizo que trabajaba par AT&T, resolvió el dilema de cuánto es necesario muestrear una señal como mínimo para poder reconstruirla luego de forma exacta a la original.

El teorema propuesto decía que como mínimo se necesita el doble de ancho de banda como frecuencia de muestreo. Esto queda reflejado de mejor manera con la siguiente expresión.

$$f_m \geq 2 BW_s$$

Hagamos un breve cálculo mental acerca de cual sería la frecuencia de muestreo para poder convertir una señal de voz humana a digital y luego poder reconstruirla en destino.

Ya habíamos dicho que para que la voz humana sea entendible es suficiente transmitir un rango de frecuencias de entre 400Hz a 4,000Hz. Por lo tanto, según el teorema de Nyquist como mínimo deberíamos muestrear al doble de la frecuencia mayor, es decir a 8,000Hz.

Luego veremos que es precisamente esa frecuencia de muestreo de 8,000Hz la que se usa en la mayoría de *codecs*. Gracias Nyquist!

1.4 Redes orientadas a circuitos

Las redes orientadas a circuitos (*circuit switched*) son aquellas donde se establece un circuito exclusivo o dedicado entre los nodos antes de que los usuarios se puedan comunicar.

Una vez que se establece un circuito entre dos puntos que quieren comunicarse, el resultado básicamente es el equivalente a conectar físicamente un par de cables de un extremo a otro. Una vez establecido el circuito, éste ya no puede ser usado por otros.

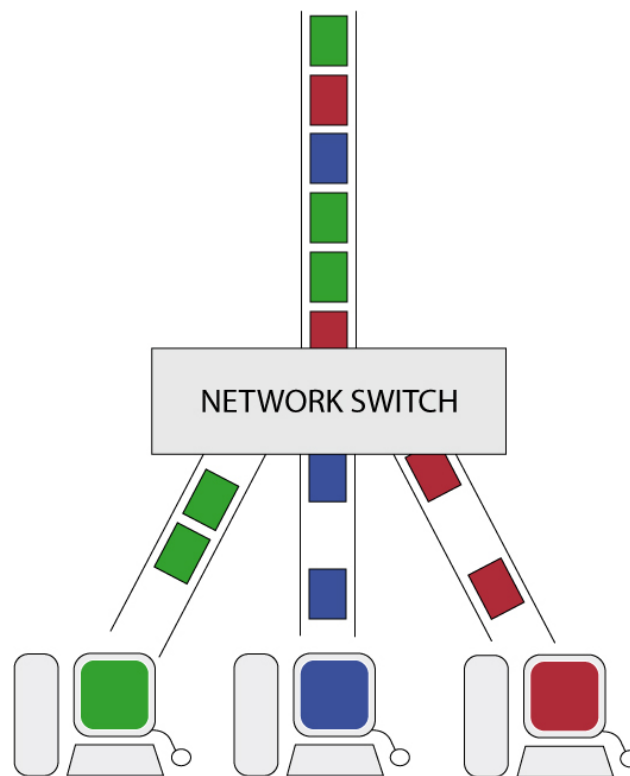
En cada circuito el retardo es constante, lo cual es una ventaja. Sin embargo, este tipo de redes es costoso debido al mismo hecho de que se necesita un circuito dedicado para cada abonado.

Este tipo de redes es el tradicionalmente usado por compañías telefónicas alrededor del mundo y es el mismo que usó Bell en sus inicios; obviamente guardando las distancias tecnológicas correspondientes.

Es común que ciertas personas confundan las redes de circuitos con las redes analógicas pero es necesario aclarar que las redes de circuitos bien pueden transportar datos digitalmente.

1.5 Redes orientadas a paquetes

Una red de paquetes es una red que por un mismo medio trafica simultáneamente diferentes flujos de información. Para hacer esto divide el tráfico de cada flujo de información en fragmentos o paquetes que envía intercaladamente. Luego, en el destino los paquetes se reensamblan para reproducir el mensaje original.



Simulación gráfica de envío de paquetes en una red de paquetes

Un ejemplo de este tipo de redes son las redes IP como es el caso del Internet, donde por una misma conexión pueden llegarnos distintos flujos de información. De esta manera podemos estar haciendo video-conferencia al mismo tiempo que enviamos un correo electrónico o navegamos por el Web. Inclusive por este tipo de redes pueden circular simultáneamente flujos de información para diferentes destinos o direcciones IP.

A diferencia de las redes orientadas a circuitos, en este tipo de redes el ancho de banda no es fijo ya que depende del tráfico de la red en un momento dado. Adicionalmente cada paquete de un mismo flujo de información no está obligado a seguir el mismo camino por lo que los paquetes que originalmente fueron generados en secuencia pueden llegar desordenados a su destino. Este tipo de factores son muy importantes a tener en cuenta cuando se trafica voz sobre una red de paquetes ya que afectan la calidad de la llamada.

Las redes de paquetes se han vuelto populares, principalmente porque optimizan recursos debido al hecho de poder utilizar el mismo medio para enviar varios flujos de información.

1.6 Red Pública Telefónica (*PSTN*)

La Red Pública Telefónica o PSTN (por sus siglas en inglés) es esencialmente una red basada en circuitos. Esta red cubre tanto telefonía fija como móvil y es la red que hace posible que podamos comunicarnos con cualquier persona en nuestra ciudad o alrededor del mundo.

Originalmente fue una red analógica pero actualmente es una red en su mayoría digital; por tanto existen dos tipos de circuitos: analógicos y digitales.

1.7 Circuitos analógicos

Los circuitos analógicos son comúnmente pares de cobre que llegan a los abonados del servicio telefónico y por donde se transmite la señal eléctrica de la voz de manera analógica. El mismo circuito lleva adicionalmente la señalización necesaria para establecer, mantener y terminar una llamada. Estos circuitos analógicos se deben conectar a un *switch* telefónico encargado de direccionar la comunicación entre los abonados.

Los circuitos analógicos están en decadencia pues las compañías telefónicas encontraron muchas ventajas en las comunicaciones digitales y es por esa razón que pese a que en la actualidad aún vemos circuitos analógicos esto se trata tan solo de la “última milla”. En cierto punto de la red telefónica esta comunicación es convertida a digital y transmitida a un *switch* telefónico digital.

La circuitería analógica comúnmente se asocia con el término de “telefonía tradicional”.

Como en el pasado era más común que los teléfonos pudieran estar ubicados en áreas rurales donde no llegaba la electricidad se decidió que la red telefónica proveyera cierto

voltaje de alimentación. Es por eso que algunos modelos de teléfonos analógicos no necesitan conectarse a la alimentación eléctrica.

En todo caso la OC (Oficina Central) genera 48 Voltios de corriente directa para alimentar a los teléfonos de los abonados. Usando léxico estricto deberíamos decir -48 Voltios debido a que este voltaje se mide con respecto a uno de los conductores. Sin embargo para ser prácticos en este libro usaremos indistintamente 48V o -48V para referirnos a lo mismo.

Señalización analógica

Para que las llamadas telefónicas funcionen correctamente es necesario contar con indicaciones o señales eléctricas que nos permitan intercambiar información entre el abonado y la OC. En breve veremos en qué consisten las señales más comunes.

Existen básicamente 3 métodos de señalización analógica que la industria ha desarrollado a través de los años. Estos se llaman *loop start*, *ground start* y *kewlstart*. Es importante cuando se configura una central telefónica que va conectada a una línea analógica que escojamos el método de señalización adecuado pues caso contrario podemos encontrarnos con problemas extraños como que la línea se cuelga inesperadamente o que no podemos colgar la línea correctamente, entre otras cosas.

La diferencia entre *loop start* y *ground start* radica en la manera en la que el teléfono requiere tono de marcado a la OC (proceso también llamado *seizure*). *Ground start* requiere tono de marcado aterrizando (de allí el término *ground*) uno de los conductores de la línea telefónica mientras que *loop start* lo hace realizando un corto circuito entre ambos conductores (es decir creando un lazo o *loop*).

Kewlstart es una evolución de *loop start* que le añade un poco más de inteligencia a la detección de desconexiones (colgado de la llamada) pero básicamente sigue siendo un *loop start*.

Debido a que *ground start* no es muy común en nuestros días, casi siempre nos veremos usando *loop start*.

A continuación explicaremos más al detalle la señalización analógica para los eventos más comunes. Para hacerlo nos basaremos en el progreso de una llamada típica usando señalización *loop start*. El progreso de una llamada lo podemos dividir en seis instancias: colgado (*on-hook*), descolgado, marcación, conmutación, ringado y conversación.

Colgado

Mientras el teléfono está colgado la OC provee un voltaje DC de 48 Voltios. El teléfono mantiene un circuito abierto con la línea telefónica; es decir que actúa como si no estuviera conectado y por lo tanto no fluye corriente por la línea.

Este estado también es conocido como *on-hook* por su significado en inglés.

Descolgado

Cuando el usuario descuelga el auricular el teléfono envía una señal a la OC. Esta señal consiste en cerrar el circuito, es decir que internamente el teléfono conecta entre sí los dos cables de la línea telefónica a través de una resistencia eléctrica.

Apenas la OC se da cuenta de esto envía tono de marcado al teléfono. Este tono de marcado le indica al abonado que ya puede marcar el número.

En gran parte de América el tono de marcado consiste en dos ondas senoidales enviadas simultáneamente. Estas ondas son de 350 Hz y 440 Hz. En Europa el tono de marcado consiste en una sola onda de 425 Hz. Sin embargo hay países en los que estos valores podrían ser diferentes.

***Nota:** Cuando era adolescente acostumbábamos a afinar la guitarra escuchando el tono de marcado del teléfono pues resulta que 440 Hz es la frecuencia de la nota musical LA que corresponde a la quinta cuerda de la guitarra.*

Marcación

La marcación puede ser por pulsos o por tonos. Los pulsos ya casi no se usan y fueron populares en los tiempos de los teléfonos de disco. Los tonos son pares de frecuencias asociadas con los dígitos telefónicos. Estas frecuencias se transmiten hasta la OC quien traduce estos tonos a números.

Más adelante se explicará más de estos tonos bajo el título DTMFs.

Conmutación

Una vez recibido los dígitos la OC tratará de asociar este número marcado con el circuito de un abonado. En caso de que el destinatario no fuere un abonado local, enviará la llamada a otro *switch* telefónico para su terminación.

Timbrado o Ringado

Una vez que la OC encuentra al abonado destino tratará de timbrarlo (*ringing*). La señal de *ring* es una onda sinusoidal de 20 Hz y de 90 Voltios de amplitud.

***Nota:** Si hemos sido observadores nos habremos dado cuenta de que la señal de ring tiene una amplitud considerable de 90 Voltios. Además recordemos que la línea ya tiene un componente adicional de Voltaje DC de 48 Voltios. Es por esta razón que si manipulamos los cables telefónicos desnudos en el preciso momento en el que llega una señal de ring podemos recibir una pequeña descarga eléctrica y pasar un buen susto.*

Adicionalmente a la señal de *ring* que la OC envía al destinatario también envía una notificación a quien originó la llamada. Este tono audible recibe el nombre de *ring-back* y consiste en dos ondas sinusoidales superpuestas de 440 Hz y 480 Hz. Estas ondas van intercaladas con espacios de silencio.

En caso de que el destinatario se encuentre ya en una llamada activa entonces en lugar del *ring-back* se devuelve un tono de ocupado a quien originó la llamada. Este tono de ocupado consiste en dos ondas sinusoidales superpuestas de 480 Hz y 620 Hz intercaladas con espacios de silencio de medio segundo.

Todos los lectores sin duda han escuchado un *ring-back* y un tono de ocupado alguna vez en sus vidas.

Conversación

Si el destinatario decide contestar la llamada el teléfono cerrará el circuito telefónico (de la misma manera que ocurrió con el teléfono que originó la llamada en la etapa de descolgado). Esta señal le informará a la OC que el destinatario decidió contestar y completará la conexión. La llamada telefónica está finalmente en curso.

DTMFs

Muchas veces es necesario enviar dígitos a través de la línea telefónica tanto para marcar como en medio de una conversación. Con esta finalidad se pensaron los DTMFs. DTMF es un acrónimo de *Dual-Tone Multi-Frequency*. Es decir que cada DTMF es en realidad dos tonos mezclados enviados simultáneamente por la línea telefónica. Esto se hace así para disminuir los errores.

A continuación una tabla ilustrando los pares de frecuencias para cada dígito.

| | 1209 Hz | 1336 Hz | 1477 Hz | 1633 Hz |
|--------|---------|---------|---------|---------|
| 697 Hz | 1 | 2 | 3 | A |
| 770 Hz | 4 | 5 | 6 | B |
| 852 Hz | 7 | 8 | 9 | C |
| 941 Hz | * | 0 | # | D |

Como se puede ver en la tabla también hay correspondencias para los signos * y # así como también para los caracteres A, B, C y D.

El teléfono analógico

Es importante hablar de este componente importante de la red telefónica pues recordemos que su invención fue lo que marcó el desarrollo del negocio de la telefonía.

Es importante también hablar del teléfono analógico porque todavía es el tipo de teléfono más común en el planeta y porque la comprensión de su funcionamiento nos permitirá entender en el futuro algunos conceptos clave como por ejemplo el eco.

En realidad el teléfono, en su forma más básica, es un dispositivo sencillo compuesto de pocos componentes.

- Auricular
- Micrófono
- Switch para colgado/descolgado
- Convertidor de dos a cuatro hilos (también llamado híbrido)
- Marcador (dialer)
- Campana o dispositivo de timbrado

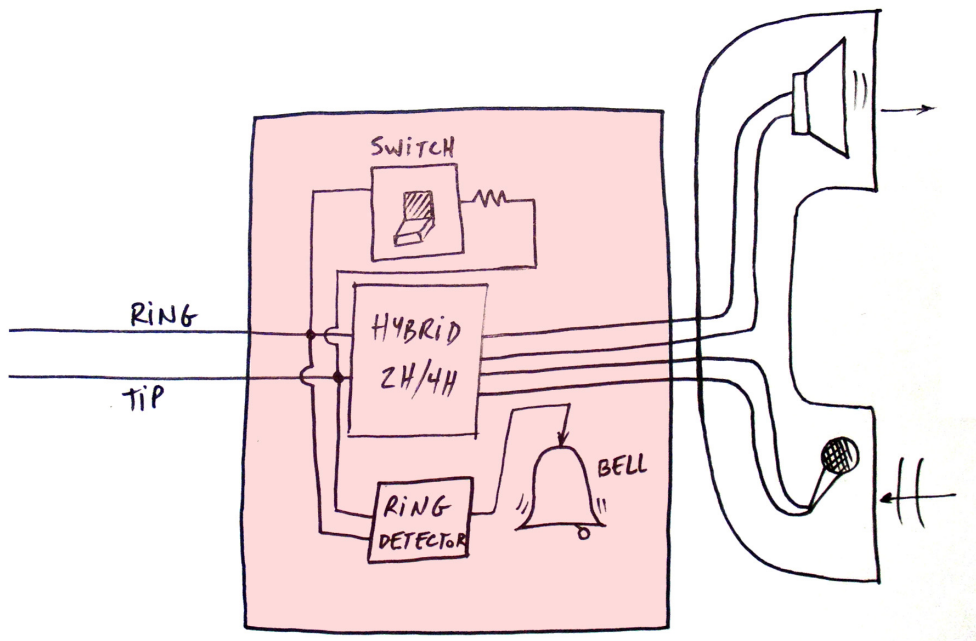


Diagrama de componentes de un teléfono

La mayoría de los componentes se explican por si solos. Sin embargo algunos se preguntarán de qué se trata el convertidor de 2 a 4 hilos?

Convertidor de 2 a 4 hilos

Un componente importante de un teléfono es el convertidor de dos a cuatro hilos, conocido también como dispositivo 2H/4H, bobina híbrida o simplemente híbrido. Este dispositivo es necesario para separar la señal de audio de ida de la de venida ya que son dos participantes en una conversación y solo existe un par de cables para esto. Si existieran tres o cuatro cables (2 de ida y 2 de venida) el convertidor de 2 a 4 hilos no fuera necesario, pero esto probablemente incrementaría los costos de cableado y las compañías telefónicas prefieren lidiar con los problemas de acoplamiento que este convertidor introduce en lugar de incrementar sus costos.

En general no existe convertidor de 2 a 4 hilos perfecto ya que es muy difícil separar las señales de ida y de regreso completamente. Es por eso que este dispositivo históricamente ha sido una de las causas de eco en líneas telefónicas mal acopladas. Ya hablaremos de esto en el capítulo de “calidad de voz”.

1.8 Circuitos digitales

La PSTN también sirve a sus abonados con circuitos digitales. Estos circuitos ofrecen la ventaja de poder multiplexar más de una línea en el mismo medio por lo que resulta atractivo para abonados con necesidades de un gran número de líneas telefónicas, por lo general empresas.

La base DS-0

Para decirlo simple, DS-0 es un canal digital de 64Kbit/s. Un DS-0 es por tanto una medida de canal estándar o unidad que nos sirve para definir múltiplos mayores como los circuitos que veremos a continuación.

Circuitos T-carrier y E-carrier

Los circuitos T-carrier (o portadora-T) fueron diseñados como nomenclatura para circuitos digitales mutiplexados y fueron desarrollados por Bell Labs hace más de cincuenta años. Los circuitos E-carrier son la equivalente europea.

El más conocido de los circuitos T-carrier es el popular T1 (y su contraparte E1). Un T1 es un circuito digital compuesto de 24 DS-0's mientras que un E1 está compuesto por 32 DS-0's. Si hacemos las matemáticas notaremos que un T1 trafica 1.544 Mbit/s mientras que un E1 2.048 Mbit/s.

Luego de los T1's tenemos múltiplos mayores como T2, T3, T4 y T5.

SONET y Circuitos Opticos

SONET (*Synchronous optical networking*) fue desarrollado con el objetivo de contar con una nomenclatura similar a las T-carrier pero usando la tecnología de fibra óptica. SONET utiliza múltiplos de T3 para sus anchos de banda y su circuito base es el llamado OC-1.

Luego del OC-1 tenemos los OC-3, OC-12, OC-24, OC-48, entre otros.

1.9 Protocolos de Señalización Digital

Los protocolos de señalización se utilizan para transmitir información de estado del canal de comunicaciones (como “desconectado”, “timbrando”, “respondido”), información de control y otra información como DTMFs, caller ID, entre otros.

Los protocolos de señalización se pueden agrupar en dos tipos llamados CAS (*Channel Associated Signaling*) y CCS (*Common Channel Signaling*). La diferencia es que mientras CAS transmite la señalización en el mismo canal en que viaja la información, CCS la transmite en un canal separado. Por este hecho es que con CAS se reduce ligeramente el ancho de banda disponible o útil para la comunicación ya que una parte de él se está usando para señalización. Esa es una de las razones por las cuales las compañías telefónicas han adoptado en su mayoría CCS.

***Nota:** No confunda el lector CAS y CCS con protocolos de señalización. Tan solo son **tipos** de protocolos que se explican aquí para hacer más fácil la categorización o agrupación de los mismos.*

Señalización Asociada al Canal (CAS)

El protocolo CAS más conocido es *robbed-bit* y es usado en circuitos T1 y E1 alrededor del mundo.

Robbed-bit toma (o “roba”, de allí su nombre) el octavo bit de cada canal de comunicación cada seis *frames* y lo reemplaza por información de señalización. El bit original robado simplemente se pierde.

Hay que notar de lo anterior que esto es posible debido a que la voz no es muy sensible que digamos a la pérdida de ese bit de información ya que es el bit menos significativo. Pero cuando transportamos data la pérdida de un bit no puede pasar desapercibida y la calidad de la transmisión se degrada de manera sensible.

Otro protocolo CAS que aún subsiste en nuestros días es R2. Se trata de un protocolo que fue popular en los años 60s. En realidad R2 es una familia de protocolos en donde cada implementación se denomina “variante”. Existen variantes dependiendo del país o inclusive de la compañía telefónica que lo ofrece.

Al momento Elastix soporta este protocolo a través de la librería Unicall. Sin embargo en el futuro se espera soportar la implementación del proyecto openR2 desarrollado por el mexicano Moisés Silva.

Señalización de Canal Común (CCS)

ISDN

ISDN (*Integrated Services Digital Network*) nos permite transmitir voz y datos simultáneamente sobre pares telefónicos de cobre con calidad superior a las líneas telefónicas analógicas.

El objetivo de ISDN fue el de facilitar las conexiones digitales para poder ofrecer una amplia gama de servicios integrados a los usuarios. ISDN establece dos tipos de interfaces para cumplir con este fin.

- BRI: Basic Rate Interface
- PRI: Primary Rate Interface

BRI estuvo orientada a hogares. Un BRI supone 2 canales útiles (también llamados canales B) de 64Kbit/s cada uno más un canal de señalización de 16Kbit/s (también llamado canal D) que en total suman 144Kbit/s.

BRI estaba llamado a ser un estándar popular en hogares pero no fue así del todo y tuvo muy poca acogida en este segmento del mercado en los Estados Unidos. En Europa la situación fue diferente y es utilizado en muchos países de este continente.

PRI es la opción para usuarios de mayor envergadura como negocios o empresas pues puede aglutinar más canales B. Actualmente es muy popular y se transmite sobre circuitos T-carrier y E-carrier.

2

Introducción a la VoIP

*Yo no hice nada por accidente, ni tampoco fueron así mis invenciones;
ellas vinieron por el trabajo*
-- Thomas Edison

La voz sobre IP o VoIP consiste en transmitir voz sobre protocolo IP.

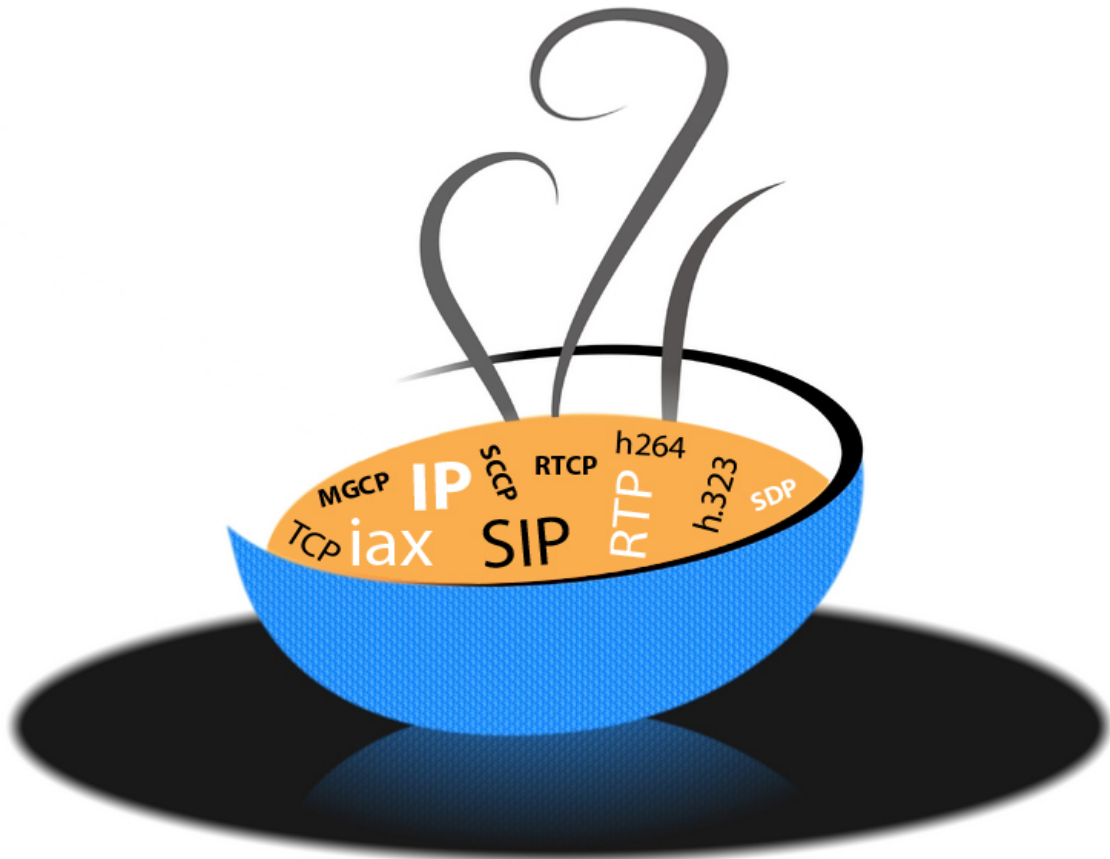
Dicho así puede sonar simple pero las redes IP fueron diseñadas principalmente para datos y muchas de las ventajas de las redes IP para los datos resultan ser una desventaja para la voz pues ésta es muy sensible a retardos y problemas de transmisión por muy pequeños que estos sean.

Por tanto transmitir voz sobre protocolo IP es toda una empresa con muchos problemas técnicos que resolver. Por suerte la tecnología ha evolucionado y la pericia de algunos ingenieros talentosos ha resultado en que podamos abstraernos en gran medida de aquellos problemas inherentes a las redes IP que perjudican la calidad de voz.

Solo hace pocos años me recuerdo haciendo uso de llamadas por Internet y puedo decir que la mejora de unos 10 años para acá ha sido notable. Ahora podemos decir que la transmisión de voz por Internet ya es una alternativa rentable al alcance de la mayoría de nosotros.

2.1 VoIP: una sopa de protocolos

Hay muchos protocolos involucrados en la transmisión de voz sobre IP. Ya de por sí hay protocolos de red involucrados como el propio protocolo IP y otros protocolos de transporte como TCP o UDP. Encima de ellos se colocan los protocolos de señalización de voz y como si esto fuera poco existen además muchas opciones de protocolos de señalización disponibles lo que puede hacer que todo suene un poco confuso al principio.



Sopa de protocolos VoIP

Clasificando los protocolos VoIP

Para simplificar las cosas podríamos clasificar a los protocolos utilizados en la VoIP en tres grupos.

Protocolos de señalización

Los protocolos de señalización en VoIP cumplen funciones similares a sus homólogos en la telefonía tradicional, es decir tareas de establecimiento de sesión, control del

progreso de la llamada, entre otras. Se encuentran en la capa 5 del modelo OSI, es decir en la capa de Sesión.

Existen algunos protocolos de señalización, que han sido desarrollados por diferentes fabricantes u organismos como la ITU o el IETF, y que se encuentran soportados por Asterisk. Algunos son:

- SIP
- IAX
- H.323
- MGCP
- SCCP

Entre estos los más populares en el ámbito de Asterisk son SIP e IAX.

Entraremos en más detalles acerca del protocolo SIP más adelante en otro capítulo.

Protocolos de transporte de voz

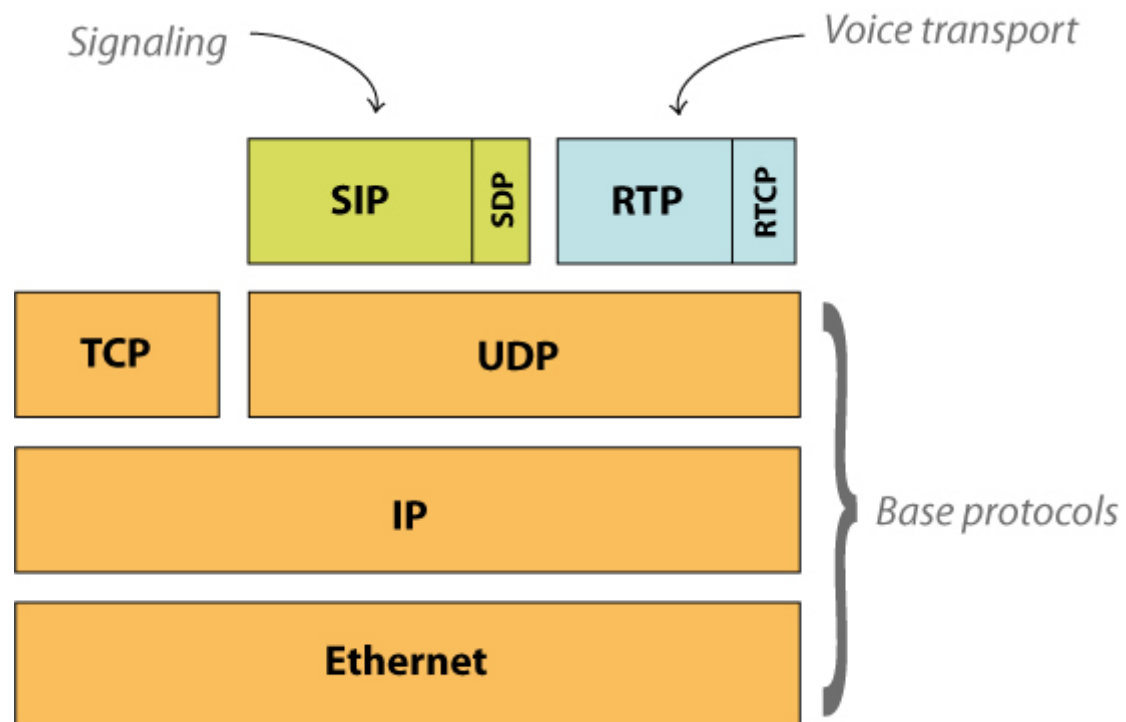
No se debe confundir aquí con protocolos de transporte de bajo nivel como TCP y UDP. Nos referimos aquí al protocolo que transporta la voz propiamente dicha o lo que comúnmente se denomina carga útil. Este protocolo se llama RTP (*Real-time Transport Protocol*) y función es simple: transportar la voz con el menor retraso posible.

Este protocolo entra a funcionar una vez que el protocolo de señalización ha establecido la llamada entre los participantes.

Protocolos de plataforma IP

En esta categoría agruparemos a los protocolos básicos en redes IP y que forman la base sobre la cual se añaden los protocolos de voz anteriores. En estos protocolos podríamos mencionar a Ethernet, IP, TCP y UDP.

***Nota:** En el presente capítulo pondremos más énfasis en explicar este grupo de protocolos con el objetivo de nivelar a quienes no tengan conocimientos sólidos en Networking. Sin embargo, los otros grupos de protocolos se revisarán con mucho más detalle en un capítulo especial dedicado a este fin.*



Protocolos involucrados en una llamada SIP. El caso de IAX es muy similar.

En la figura anterior podemos observar un hecho curioso y es que pese a que SIP soporta tanto UDP como TCP sólo lo vemos posado sobre UDP. No se trata de un error sino más bien que en Asterisk la implementación de SIP solo está disponible para UDP.

2.2 Protocolo IP

El protocolo IP sin duda es uno de los más populares jamás implementados, principalmente por el auge del Internet: La gran red de redes, que utiliza este protocolo para su enrutamiento. Por esta razón, antes de entrar en detalles acerca del protocolo IP emplearemos un par de párrafos a resumir los orígenes del Internet.

El Internet

El Internet tuvo su origen en la década del 60 como un proyecto del Departamento de Defensa de los Estados Unidos con el objetivo de crear una red robusta que conectara algunos organismos públicos con universidades. La idea de la red, que originalmente se llamó ARPANet, era que si alguno de los nodos quedaba fuera de funcionamiento esto no perjudicara significativamente a la red. Para esto se ideó una red descentralizada que pudiera manejar más de un camino entre dos de sus puntos. En resumen, se trataba de construir una red a prueba de fallos (recordemos también que eran épocas de la Guerra Fría) y de bajo coste.

Gracias a este enfoque ARPANet evolucionó hasta lo que hoy en día conocemos como Internet, una red de comunicaciones tan grande y poderosa que sin duda ha ejercido un impacto en nuestras culturas. Sin su existencia el proyecto Elastix no hubiera existido y no estarían leyendo esto ahora.

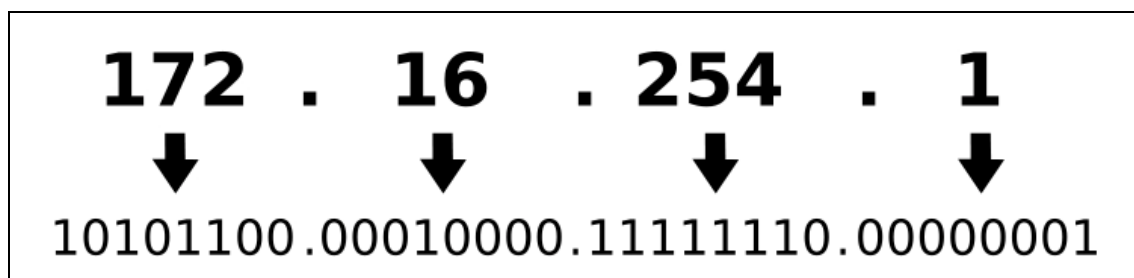
Qué es el protocolo IP?

El protocolo IP (*Internet Protocol*) es un protocolo que trabaja a nivel de red donde la información se envía en paquetes llamados paquetes IP. Este protocolo ofrece un servicio “sin garantías” también llamado del “mejor esfuerzo”. Es decir que nada garantiza que los paquetes lleguen a destino, sin embargo se hará lo posible por hacerlos llegar.

Dirección IP

Una dirección IP es un número único que provee además información de cómo ubicar al equipo que la posee. Para que este número sea único existe una autoridad que controla la asignación de direcciones IP llamada IANA (*Internet Assigned Numbers Authority*).

Una dirección IP es un número binario que consta de 32 bits. Sin embargo, para fines prácticos y hacer que este número sea más entendible para los humanos casi siempre se representa en un formato de 4 números decimales separados por puntos. Cada uno de estos cuatro números puede tomar un valor de 0 a 255. Un ejemplo de dirección IP es 172.16.254.1



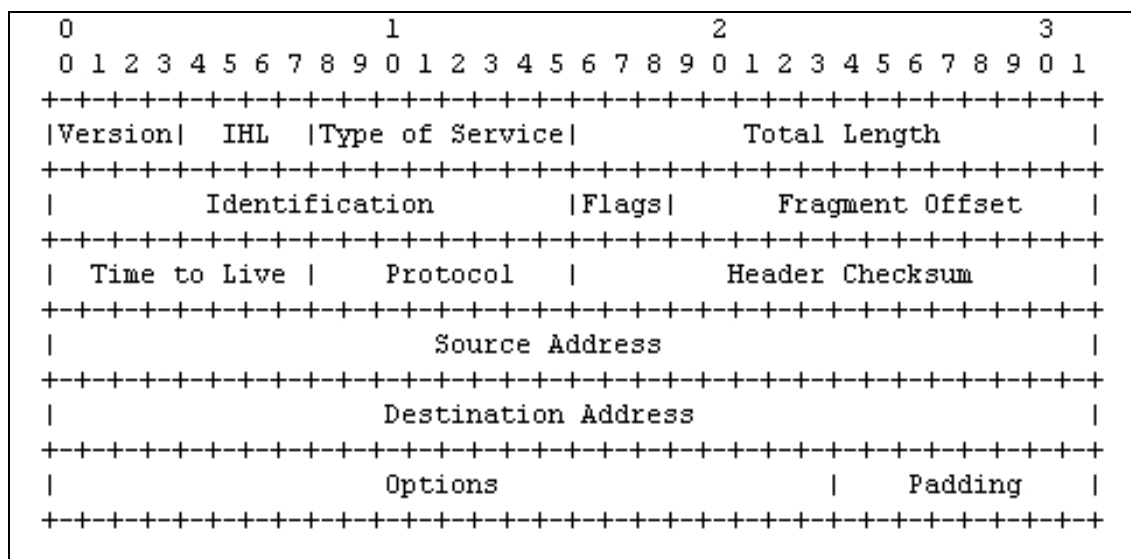
Traducción de notación decimal a binaria (32 bits) de una dirección IP

Paquete IP

Como habíamos dicho antes el protocolo IP es un protocolo que divide la información en paquetes que envía a su destino y la ventaja de tener la información paquetizada es que estos paquetes pueden tomar diferentes caminos para llegar a destino. Es decir que hay redundancia de caminos y es menos probable que todos los paquetes se pierdan.

Bueno, bueno, pero... cómo lucen los famosos paquetes?

El formato de un paquete IP está diseñado para llevar información que permita diseccionarlo a su destino y obviamente que permita re-ensamblar los paquetes en destino para recuperar la información útil. A continuación una figura con una cabecera de paquete IP.



Cabecera de un paquete IP

Direccionamiento IP

El direccionamiento o (enrutamiento) permite determinar la ruta óptima para que un paquete IP llegue a su destino. Para saber dónde quiere llegar un paquete dado hay que examinar la dirección IP de destino en el campo *Destination Address* de la cabecera de un paquete IP.

Los equipos que se encargan de enrutar los paquetes a su destino se llaman routers y básicamente contienen tablas de rutas con información de cómo alcanzar otras redes. Por tanto, una vez que llega un paquete a un router éste examina la dirección IP destino y trata de determinar a qué red pertenece esa dirección IP.

Determinar la red donde quiere llegar un paquete dado no es muy difícil puesto que esa información es parte de la dirección IP. Esto es porque cada dirección IP se podría dividir en dos partes: una parte que identifica a la red y otra que identifica al equipo (también llamado *host*).

Identificar cada una de estas partes no es tan trivial pues existen redes de diferentes tamaños y por tanto el número de bits que corresponden a la parte de la red y el número de bits que corresponden a la parte del *host* varían dependiendo del tamaño de la red y se regulan mediante un parámetro adicional llamado máscara de red.

Máscara de red y cálculo de dirección de red

Veamos entonces cómo se obtiene la información de a qué red pertenece una dirección IP para que un ruteador pueda encaminar correctamente un paquete a destino.

La máscara de red es un número de 32 bits al igual que una dirección IP. Se llama máscara porque si se superpone a la dirección IP nos permite identificar cuál parte es la que corresponde a la dirección de red y cuál a la dirección del host.

Imaginemos que tenemos la dirección IP 130.5.5.26 con máscara 255.255.255.0. Para ilustrar cómo aplicar la máscara sobre la dirección IP convirtamos estos datos a binario.

| IP Address | | Network | Host |
|---------------|----|----------------------------|----------|
| 130.5.5.26 | -> | 10000010.00000101.00000101 | 00011010 |
| 255.255.255.0 | -> | 11111111.11111111.11111111 | 00000000 |
| Mask | | | |

Aplicación de máscara para obtener información de la red y *host*

Ahora podemos ver con más claridad que los números uno (1) en la máscara marcan el límite entre la parte de la red y la parte del *host* por lo tanto la dirección de red es (en binario):

10000010.00000101.00000101.00000000

Todos los números binarios a la derecha de la máscara se completan con ceros. Ah cierto, olvidamos volver a convertir la dirección de red anterior a decimal. Veamos qué obtenemos:

10000010.00000101.00000101.00000000 → 130.5.5.0

Lo que quiere decir que la dirección IP 130.5.5.26 pertenece a la red 130.5.5.0. Ahora el ruteador ya puede decidir hacia qué red enviar nuestro paquete IP.

2.3 Protocolos de transporte

***Nota:** No se debe confundir estos protocolos con el “protocolo de transporte de voz” RTP. RTP es un protocolo más especializado que incluso se monta sobre UDP (un protocolo que será explicado aquí). Los protocolos de transporte aquí cubiertos son de propósito general y reposan directamente sobre el protocolo IP.*

Protocolo TCP

Como ya habíamos dicho el protocolo IP no garantiza que los datos lleguen a destino. Solo hace su mejor esfuerzo para que lleguen.

Por lo tanto era necesario un protocolo que se encargue de controlar la transmisión de datos y por esta razón se diseñó lo que se llama *Transmission Control Protocol* o simplemente protocolo TCP. TCP es un protocolo de transporte que se transmite sobre IP.

TCP ayuda controlando que los datos transmitidos se encuentren libre de errores y sean recibidos por las aplicaciones en el mismo orden en que fueron enviados. Si se pierden datos en el camino introduce mecanismos para que estos datos sean reenviados.

Obviamente esto implica una carga extra de información en el flujo de datos ya que hay que enviar información de control adicional. Es por esto que TCP es un buen protocolo para control de sesiones pero no tan bueno para transmisión de datos en tiempo real. Por esta razón la voz en sí no se envía usando este protocolo. Sin embargo TCP juega un rol muy importante en muchos protocolos relacionados con un servidor Elastix.

TCP es quien introduce el concepto de “puerto” que no es otra cosa que una abstracción para poder relacionar los flujos de datos con servicios de red específicos (o protocolos de más alto nivel). Por ejemplo, el puerto 80 se asocia con el servicio de Web o el protocolo HTTP; el puerto 25 se asocia con el servicio de correo electrónico o protocolo SMTP.

Protocolo UDP

UDP (*User Datagram Protocol*) es otro protocolo de transporte. Se diferencia con TCP en que a este protocolo no le importa si los datos llegan con errores o no y tampoco le importa si llegan en secuencia. La pregunta entonces es para qué sirve este protocolo?

Bueno, un protocolo de transporte no necesariamente tiene que garantizar que la información llegue a destino o llegue en secuencia. Esta es solo una característica extra. Es más o menos análogo a un servicio de transporte de mercancía. Imaginémonos una flota de motocicletas que ofrece el servicio de transporte en una ciudad e imaginémonos que le encomendamos a dicha flota la tarea de transportar un gran cargamento de

archivos de una oficina u otra. La compañía se encargará de dividir nuestro cargamento de archivos y distribuirlo en cantidades o paquetes que puedan ser transportados en sus vehículos. Hará lo necesario para que nuestro cargamento llegue a destino. Esto es en esencia el servicio de transporte. Sin embargo podemos escoger dos clases de servicio: una que garantiza que la mercancía llegue segura y otra que no. Estos dos tipos de servicio son análogos a los dos tipos de protocolos de transporte que estamos describiendo en este apartado UDP y TCP.

En fin, la cosa es que UDP divide la información en paquetes, también llamados datagramas, para ser transportados dentro de los paquetes IP a su destino.

Al no ser necesario incluir mucha información de control, el protocolo UDP reduce la cantidad de información extra en los paquetes por lo que es un protocolo más rápido que TCP y adecuado para transmisión de información que debe ser transmitida en tiempo real como la voz.

Es por esta razón que la voz en aplicaciones de VoIP es transmitida sobre este protocolo.

2.4 Codificación de la voz

Ya tenemos claro que para transportar la voz se utilizan algunos protocolos como SIP, IAX y otros como RTP o RTCP. Pero la voz es una onda analógica que necesita transformarse a digital en algún formato antes de ser transmitida.

Lógicamente podríamos tratar de transmitirla tal cual resulta de la conversión analógica-digital (ADC) pero resulta que nos encontramos en una red de paquetes así que debemos paquetizar esta información. Además si la transmitimos tal cual resulta de la conversión ADC desperdiciaríamos recursos de la red por lo que hace falta encontrar un formato óptimo.

Esa búsqueda de un formato óptimo generó algunas alternativas de formatos de transmisión llamadas *codecs*.

***Nota:** Los codecs realmente no son exclusivos de la VoIP pues también se usan en otros tipos de comunicaciones digitales.*

Codecs

La palabra *codec* proviene de abreviar las palabras **C**odificación y **D**Ecodificación. Su función principal es la de adaptar la información digital de la voz para obtener algún

beneficio. Este beneficio en muchos casos es la compresión de la voz de tal manera que podamos utilizar menos ancho de banda del necesario.

Algunos *codecs*, soportados por Asterisk y comúnmente usados en comunicaciones de VoIP, son G.711, G.729, GSM, iLBC, entre otros.

Explicaremos brevemente tres de ellos.

G.711

G.711 es uno de los *codecs* más usados de todos los tiempos y proviene de un estándar ITU-T que fue liberado en 1972. Viene en dos sabores llamados u-law y a-law. La primera versión se utiliza en los Estados Unidos y la segunda se utiliza en Europa.

Una de sus características es la calidad de voz debido a que casi no la comprime. Utiliza 64kbit/s, es decir un muestreo de 8 bits a 8kHz. Es el *codec* recomendado para redes LAN pero hay que pensarlo dos veces antes de utilizarlo en enlaces remotos debido al alto consumo de ancho de banda.

El soporte para este *codec* ya viene habilitado en Elastix.

G.729

También se trata de una recomendación ITU cuyas implementaciones ha sido históricamente licenciadas, o sea que hay que pagar por ellas.

La ventaja en la utilización de G.729 radica principalmente en su alta compresión y por ende bajo consumo de ancho de banda lo que lo hace atractivo para comunicaciones por Internet. Pese a su alta compresión no deteriora la calidad de voz significativamente y por esta razón ha sido ampliamente usado a través de los años por muchos fabricantes de productos de VoIP.

G.729 utiliza 8kbit/s por cada canal. Si comparamos este valor con el de G.711 notaremos que consume 8 veces menos ancho de banda, lo cual a simple vista es un ahorro de recursos significativo.

Nota: Existen variaciones de G.729 que utilizan 6.4kbit/s y 11.8kbit/s.

Para habilitar canales G.729 en Elastix hay que comprar una licencia por cada canal. Esto se puede adquirir en el sitio Web de Digium.

GSM

Muchas personas suelen preguntar si el *codec* GSM tiene algo que ver con el estándar de comunicaciones celulares y la respuesta es que sí.

El estándar que define la tecnología celular GSM (*Global System for Mobile communications*) incluye este *codec*.

La ventaja de este *codec* también es su compresión. Acerca de la calidad de voz... bueno, ya habremos hablado por un celular GSM alguna vez.

GSM comprime aproximadamente a 13kbit/s y ya viene habilitado en Elastix.

2.5 Sobrecarga de protocolos

Como ya vimos, para transportar la voz de un lugar a otro, en una red de paquetes, necesitamos la ayuda de algunos protocolos; pero ya nos habremos dado cuenta de que estos protocolos transmiten data adicional que ocupa ancho de banda extra a la voz propiamente dicha. Algunos de ellos son Ethernet, IP, UDP, RTP.

En resumen esto hace que el ancho de banda real para transmitir voz sea mayor al del *codec*.

Por ejemplo, para transmitir voz usando G.711 en teoría deberíamos usar 64Kbps (peso del *codec*) pero en realidad usamos 95.2Kbps de BW. En otros *codecs* más compresores la sobrecarga es incluso más significativa (porcentualmente hablando).

Hagamos un ejercicio sencillo usando como ejemplo G.711. Veamos la siguiente figura para ilustrar mejor el ejercicio.

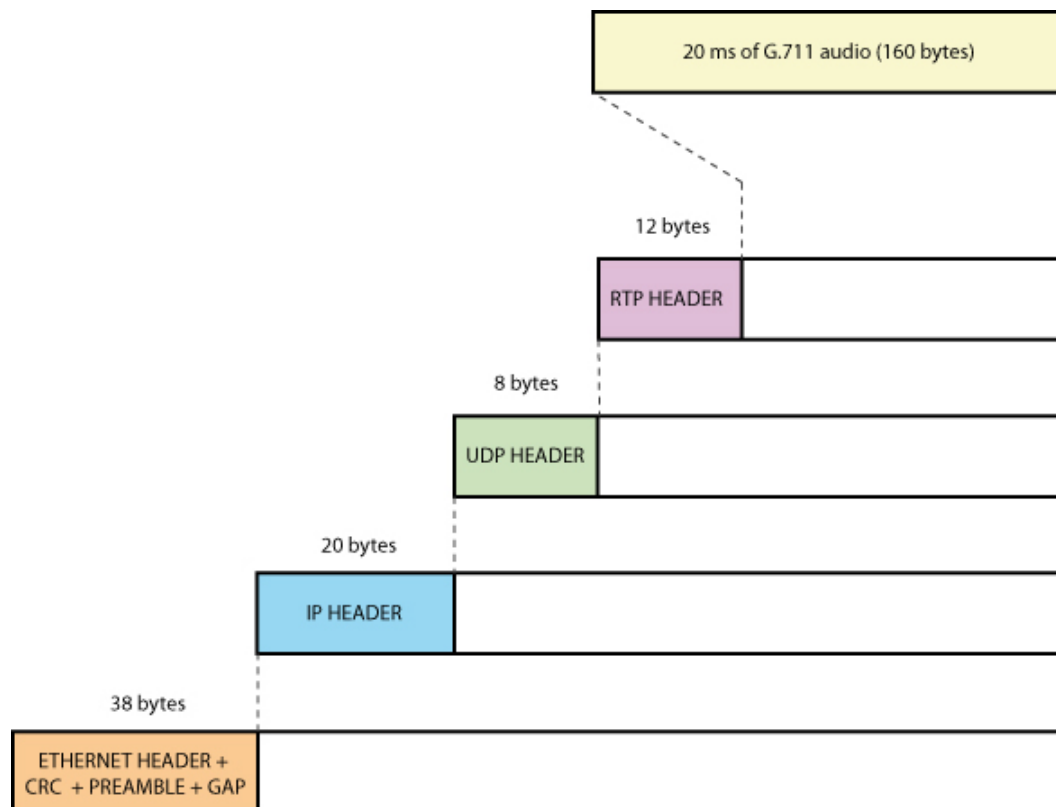


Ilustración de la sobrecarga de protocolos usando G.711

Calculemos el ancho de banda para G.711 basándonos en la figura anterior, la cual nos muestra el desperdicio de cada protocolo. Sumemos ahora todos estos bytes enviados.

Bytes transmitidos cada 20ms = $38 + 20 + 12 + 8 + 160 = 238$ bytes

Bits transmitidos cada 20ms = $238 \text{ bytes} * 8 \text{ bits/byte} = 1904$ bits

Bits transmitidos cada segundo = $1904 \text{ bits/frame} * 50 \text{ frames/seg.} = 95,200$ bits/segundo = **95.2Kbps!**

Comparativa de *codecs*

A continuación una tabla que muestra el *overhead* para algunos de los *codecs* más populares soportados por Asterisk.

| Codec | Ancho de banda códec | Ancho de banda real Ethernet | Porcentaje de overhead |
|--------|----------------------|------------------------------|------------------------|
| G.711 | 64 Kbps | 95.2 Kbps | 48.75% |
| iLBC | 15.2 Kbps | 46.4 Kbps | 205.26% |
| G.729A | 8 Kbps | 39.2 Kbps | 390% |

Si el lector está interesado en cálculos de ancho de banda real para otros codecs, le recomendamos el siguiente vínculo: <http://www.newport-networks.com/pages/voip-bandwidth-calculator.html>

3

Linux para Administradores de Elastix

Sólo los débiles hacen copias de seguridad en cintas: los hombres de verdad tan solo suben sus cosas importantes a un servidor ftp, y dejan que el resto del mundo las replique.
-- Linus Torvalds, creador de Linux

3.1 Introducción

En este capítulo se repasarán algunos conceptos útiles de Linux para facilitar la vida de los administradores de Elastix. Si usted ya conoce cómo administrar su Elastix desde la línea de comandos de Linux, sáltese este capítulo y vaya directamente al siguiente.

Debido a que este no es un libro de Linux no ahondaremos mucho en el tema, ni hablaremos de la historia de Linux, ni de sus ventajas, ni ahondaremos en detalles. Así que si en algún momento le parece que estoy abordando los temas de manera muy general... esa es la idea :)

Si al final de este capítulo aún siente que existen vacíos, le recomendamos capacitarse en administración de Linux, sin duda le facilitará las tareas de administración de Elastix.

3.2 Línea de comandos de Linux

Una manera muy útil de administrar Elastix es desde la línea de comandos de Linux. La línea de comandos de Linux nos permitirá explotar al máximo el potencial de nuestro Servidor de Comunicaciones Unificadas.

Para ingresar a la línea de comandos lo podemos hacer directamente conectando un monitor y teclado a nuestro equipo instalado con Elastix o conectándonos remotamente a través del protocolo SSH.

Si estamos en un equipo Windows debemos descargar un programa cliente para podernos conectar remotamente. El programa que recomendamos aquí se llama PuTTY y basta hacer una búsqueda en Google por ese nombre para encontrar el sitio de descargas. Sólo asegúrese de hacerlo del sitio oficial.

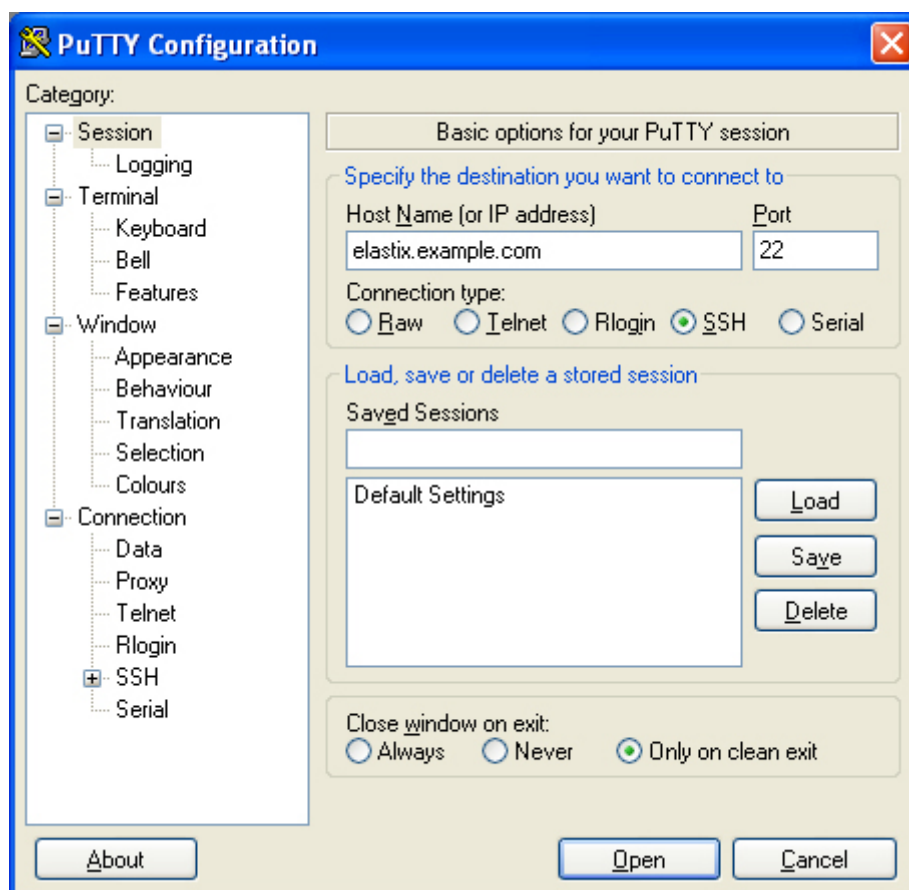


Figura de la aplicación PuTTY

```

edgar@localhost:~
login as: edgar
edgar@192.168.0.10's password:
Last login: Fri Sep 19 03:45:18 2008 from 192.168.0.1
[edgar@localhost ~]$ ls -la
total 1981064
drwx----- 64 edgar edgar      4096 2008-09-19 04:51 .
drwxr-xr-x  3 root  root      4096 2008-07-23 12:16 ..
-rw-rw-r--  1 edgar edgar 1806310955 2008-09-15 00:25 100_0870.MOV
drwx-----  5 edgar edgar      4096 2008-07-30 14:02 .adobe
drwxrwxr-x  2 edgar edgar      4096 2008-08-11 18:17 .assistant
drwxrwxr-x  2 edgar edgar      4096 2008-08-04 01:42 autosave
-rw-----  1 edgar edgar    25687 2008-09-19 04:51 .bash_history
-rw-r--r--  1 edgar edgar      18 2008-02-29 09:27 .bash_logout
-rw-r--r--  1 edgar edgar     176 2008-02-29 09:27 .bash_profile
-rw-r--r--  1 edgar edgar     124 2008-02-29 09:27 .bashrc
drwxrwxr-x  3 edgar edgar      4096 2008-09-01 02:22 .ccache
drwxr-xr-x  5 edgar edgar      4096 2008-08-08 20:19 .config
drwxr-xr-x  2 edgar edgar      4096 2008-07-23 21:18 .dasher
drwx-----  3 edgar edgar      4096 2008-07-23 21:03 .dbus
drwxr-xr-x  6 edgar edgar      4096 2008-09-19 04:42 Desktop
drwxr-xr-x  2 edgar edgar      4096 2008-09-18 03:02 Documents
drwxr-xr-x 14 edgar edgar      4096 2008-09-19 04:21 Download
drwxrwxr-x  3 edgar edgar      4096 2008-08-28 00:22 .eclipse
drwxrwxr-x  3 edgar edgar      4096 2008-08-29 16:48 elastix

```

Figura ejecutando PuTTY

3.3 Comandos básicos

A continuación una lista de los comandos básicos más usados:

| Comando | Descripción |
|---------------------------------|---|
| ls | Lista los archivos en el directorio actual |
| ls -la | Lista los archivos y sus atributos |
| cd | Cambia de directorio |
| cd .. | Cambia al directorio padre |
| cat | Muestra el contenido de un archivo en la pantalla |
| pwd | Muestra la ruta actual |
| cp | Copia un archivo o grupo de ellos |
| mv | Mueve un archivo. También se lo usa para renombrar archivos |
| rm | Elimina un archivo |
| tail -f nombre_archivo | Muestra las últimas líneas de un archivo |
| mkdir | Crea un directorio |
| tar -xvzf nombre_archivo.tar.gz | Descomprime un archivo tar.gz |
| top | Muestra un reporte en tiempo real de los |

| | |
|-----------------|--|
| | procesos que se ejecutan en el sistema. Puede ser ordenado por consumo de memoria o CPU. Muy útil para diagnóstico |
| find | Permite buscar archivos |
| vim | Programa para ver y editar archivos de texto plano |
| ifconfig | Muestra información de las interfaces de red y permite modificarlas temporalmente |
| ps -aux | Muestra un reporte de los procesos que se están ejecutando en el servidor |
| reboot | Reincia el equipo |
| shutdown -h now | Apaga el equipo |

```

root@elastix:~
File Edit View Terminal Tabs Help
top - 04:43:48 up 8 days, 20:18, 1 user, load average: 0.00, 0.01, 0.00
Tasks: 136 total, 1 running, 134 sleeping, 0 stopped, 1 zombie
Cpu(s): 0.2%us, 0.2%sy, 0.0%ni, 90.9%id, 0.0%wa, 8.8%hi, 0.0%si, 0.0%st
Mem: 1026444k total, 988228k used, 38216k free, 76392k buffers
Swap: 779144k total, 200k used, 778944k free, 394988k cached

  PID USER      PR  NI  VIRT  RES  SHR  S  %CPU  %MEM    TIME+  COMMAND
 18152 daemon    21   0 310m 181m 7920  S   0 18.1  925:37.52 java
   2406 mysql     15   0 152m  38m 5284  S   0  3.9  328:35.40 mysqld
   4890 root      24   0 43440  37m 2332  S   0  3.8   1:01.05 spamd
 10510 root      15   0 36552  31m 2316  S   0  3.1   0:08.53 spamd
   2610 root      15   0 34532  29m 2408  S   0  2.9   0:41.37 spamd
   4200 asterisk  15   0 40212  21m 6292  S   0  2.1   0:39.12 httpd
   7729 asterisk  16   0 39460  19m 5552  S   0  2.0   0:22.51 httpd
   4144 asterisk  16   0 38288  19m 6236  S   0  1.9   0:41.52 httpd
   4112 asterisk  18   0 38372  19m 6196  S   0  1.9   0:37.93 httpd
   4744 asterisk  18   0 38100  19m 6060  S   0  1.9   0:28.40 httpd
   4198 asterisk  17   0 37900  18m 6076  S   0  1.9   0:35.69 httpd
 10400 asterisk  16   0 38508  18m 5448  S   0  1.9   0:11.84 httpd
   4199 asterisk  18   0 36956  17m 6084  S   0  1.8   0:35.75 httpd
   2245 asterisk  19   0 37324  16m 6832  S   0  1.7  42:16.44 asterisk
 10409 asterisk  17   0 33284  13m 5396  S   0  1.4   0:11.36 httpd
 10835 asterisk  16   0 33216  13m 5396  S   0  1.4   0:06.59 httpd
 10408 asterisk  15   0 33196  13m 5332  S   0  1.3   0:10.70 httpd
 10834 asterisk  15   0 33148  13m 5280  S   0  1.3   0:07.10 httpd
 10410 asterisk  16   0 32708  13m 5284  S   0  1.3   0:10.83 httpd
 10833 asterisk  15   0 32412  12m 4576  S   0  1.2   0:06.84 httpd
 14792 root      18   0 25520  9316 6228  S   0  0.9   0:04.90 httpd
 24614 asterisk  15   0 12864  9136 1872  S   0  0.9   1:58.02 op_server.pl
   2286 ntp       15   0  4332  4332 3324  S   0  0.4   0:06.34 ntpd
   2008 root      15  -3 10112  3868 2280  S   0  0.4   0:02.97 python
   2865 haldaemo 18   0  5488  3668 1644  S   0  0.4   0:01.30 hald

```

Figura del la ejecución del comando top

3.4 Sistema de archivos

En Linux es muy importante la organización del sistema de archivos.

Esta organización le otorga a Linux robustez e interoperabilidad pues los programas pueden predecir con mayor exactitud dónde se encuentran ciertos archivos para su correcto funcionamiento y así mismo escribir archivos de manera que otros componentes los entiendan. Esto minimiza los conflictos y en conjunto con el sistema de permisos añade más seguridad al sistema operativo.

Organización

La organización de directorios que se mostrará a continuación no es arbitraria. Es producto de años de evolución desde las primeras versiones de UNIX.

Esta estructura de carpetas está ubicada en lo que se llama directorio raíz, o directorio principal. Este directorio raíz es el directorio de más alto nivel en el sistema de archivos.

| Directorio | Descripción |
|-------------------|---|
| <code>sbin</code> | Incluye binarios que pueden ser ejecutados solo por el administrador |
| <code>bin</code> | Contiene binarios necesarios para el correcto funcionamiento del sistema. Los binarios de esta carpeta pueden también ser utilizados por los usuarios del sistema |
| <code>boot</code> | Este directorio contiene el kernel y otros archivos necesarios al momento de arranque |
| <code>dev</code> | Este directorio contiene archivos que representan dispositivos de hardware. Recordemos que en Linux prácticamente todo es un archivo |
| <code>etc</code> | Este directorio contiene básicamente archivos de configuración para todo el sistema |
| <code>home</code> | Aquí residen los directorios de los usuarios. Por ejemplo, el directorio <code>/home/jorge</code> es el directorio del usuario “jorge” |
| <code>lib</code> | Contiene básicamente las librerías compartidas que requieren los programas. Algo análogo a los archivos <code>.dll</code> en Windows |

| | |
|------------|---|
| lost+found | Cuando por alguna razón el sistema de archivos se corrompe y hay que repararlo; los resultados de esa reparación se guardan en este directorio |
| mnt | Este es un lugar común para montar otros sistemas de archivos o dispositivos. Por ejemplo una unidad de CDROM o un PEN DRIVE |
| opt | Este directorio contiene paquetes de software que normalmente no son parte de la instalación original |
| proc | Es un directorio virtual (ya que realmente reside en la memoria) donde el kernel y drivers escriben archivos para que otros programas los puedan leer. Por ejemplo, aquí es donde los drivers de zaptel escriben ciertos archivos en la ruta /proc/zaptel/ |
| root | Este es el directorio del usuario root ya que este no tiene uno del tipo /home/root |
| tmp | Este directorio contiene archivos temporales |
| usr | Es uno de los más importantes directorios y contiene información para ser compartida como programas que pueden ser usados por los usuarios del sistema, la documentación de dichos programas y hasta librerías relacionadas, etc. Este directorio usualmente no contiene información que cambie constantemente |
| var | Contiene información en constante cambio como logs o colas de emails. Por ejemplo, aquí reside el log de asterisk en la ruta /var/log/asterisk/full |

Permisos

En Linux cada directorio o archivo tiene permisos de acceso y estos permisos pueden ser de básicamente 3 tipos:

- lectura
- escritura
- ejecución

Cada archivo también tiene un dueño y un grupo al cual pertenece y para cada una de estas entidades se pueden establecer permisos diferentes.

Por último, también es deseable establecer permisos para otros usuarios que eventualmente pueden querer tener acceso sobre este archivo.

Dicho todo lo anterior podemos establecer la siguiente matriz de permisos que sería válida para un determinado archivo del sistema:

| | dueño | grupo | otros usuarios |
|-----------|-------|-------|----------------|
| lectura | x | x | x |
| escritura | x | | |
| ejecución | x | x | x |

Existe sin embargo una notación más sencilla para expresar la matriz de arriba. Esta notación, compuesta de 10 caracteres, simplifica el despliegue de la información de permisos y permite mostrarlos en una sola línea lo cual es muy útil cuando se quieren ver permisos de muchos archivos. Por ejemplo, la matriz de permisos del ejemplo anterior se puede expresar de la siguiente manera:

-rwxr-xr-x
0123456789

Donde,

- El carácter 0 representa el tipo de archivo. Si se llena con la letra “d” significa que se trata de un directorio.
- Los caracteres 1, 2 y 3 representan los permisos de lectura, escritura y ejecución para el dueño del archivo.
- Los caracteres 4, 5 y 6 representan los permisos de lectura, escritura y ejecución para el grupo al que pertenece el archivo.
- Los caracteres 7, 8 y 9 representan los permisos de lectura, escritura y ejecución para otros usuarios.

Ahora veamos un ejemplo de un listado largo donde podremos apreciar diferentes permisos:

```
$ ls -la
```

```
drwxrwxr-x 4 edgar edgar      4096 2008-09-01 21:20 Codigo
-rw-rw-r-- 1 edgar edgar      53058 2008-08-04 01:31 comohemo.mid
-rw-rw-r-- 1 edgar edgar    22785270 2008-08-07 12:09 dlj173.pdf
drwxrwxr-x 3 root  root        4096 2008-08-26 17:21 elastix
drwxrwxr-x 2 test  prueba      4096 2008-09-11 13:03 flashingfop
drwxrwxr-x 3 edgar edgar      4096 2008-08-04 22:18 fpbx
-rw-rw-r-- 1 edgar edgar    398439 2008-08-19 19:19 gforge_manual.pdf
-rwxrw-r-- 1 edgar edgar     102135 2008-09-07 18:53 index2.html
-rw-rw-r-- 1 user4 101         3535 2008-08-19 23:20 install-ceros
drwxrwxr-x 3 edgar edgar      4096 2008-08-03 01:52 installers
-rw-rw-r-- 1 edgar edgar      6860 2008-08-19 23:12 ksraid.cfg
drwxr-xr-x 4 edgar edgar      4096 2008-08-06 12:14 lw
-rw-rw-r-- 1 edgar edgar    1020152 2008-07-30 14:00 Magic Button.pdf
drwxrwxr-x 3 edgar edgar      4096 2008-09-02 17:25 monitoring
-rw-rw-r-- 1 edgar edgar    14341913 2008-08-12 01:59 prbx1.zip
-rw-rw-r-- 1 edgar edgar     7195354 2008-08-03 01:51 rotron1.7.1.tar.bz2
-rw-rw-r-- 1 edgar edgar     25343 2008-08-13 00:25 roundcube.diff
-rw-rw-r-- 1 edgar edgar     12167 2008-08-01 23:03 src-php.tar.bz2
drwxr-xr-x 18 edgar edgar      4096 2008-08-13 09:38 wanpipe-3.2.7
-rw-rw-r-- 1 edgar edgar    16106184 2008-08-29 17:40 wanpipe-3.2.7.1.tgz
-rw-rw-r-- 1 edgar edgar     20525 2008-09-08 17:14 xorcom_test.txt
-rw-rw-r-- 1 edgar edgar     9277 2008-09-08 02:10 Xorcom_logo.png
```

Representación numérica de permisos

Aparentemente vamos a complicar las cosas un poco más aquí pero ya veremos más adelante la utilidad de poder representar la dichosa matriz de permisos de una forma más condensada llamada representación numérica u octal de permisos.

La cuestión es que los 3 caracteres que definen los permisos para cada una de las tres entidades involucradas (dueño, grupo y otros) pueden ser representados por un solo número de tal manera que todos los permisos de un archivo pueden ser representados por 3 de estos números. Por ejemplo, 755. Donde el primer dígito representa los permisos del dueño, el segundo dígito los permisos del grupo y el tercer dígito los permisos de otros usuarios.

Puede ser complicado entender al principio pero sin duda se trata de un formato más condensado. De eso sí no cabe duda verdad?. Y vasta ver el siguiente ejemplo:

`-rwxr-xr-x` equivale a 755

Pero de dónde sale el famoso numerito?

Resulta que cada uno de los 3 caracteres que representan un permiso para una entidad dada puede ser reemplazado por 1 o 0 de tal manera que obtenemos un número binario, luego este número binario se traduce a su equivalente decimal (realmente deberíamos decir octal aquí). Veamos el ejemplo siguiente.

`-rwxr-xr-x`

Transformado a binario tendríamos los siguientes tríos:

111 101 101

y ahora en su equivalente decimal

755

En fin, para los que no les gusta calcular binario les dejo la siguiente tablita.

| Versión texto | Versión binaria | Versión Octal |
|---------------|-----------------|---------------|
| --- | 000 | 0 |
| --x | 001 | 1 |
| -w- | 010 | 2 |
| -wx | 011 | 3 |
| r-- | 100 | 4 |
| r-x | 101 | 5 |
| rw- | 110 | 6 |
| rwX | 111 | 7 |

Ahora sí se estarán preguntando para qué nos sirve esta notación numérica?

Bueno, resulta que de esta manera es mucho más fácil manipular los permisos de los archivos mediante comandos y eso es precisamente lo que veremos a continuación.

Cambiando permisos y dueños de archivos

Las tareas de cambio de permisos y dueños de archivos pueden ser realizadas fácilmente gracias a los comandos **chmod** y **chown** respectivamente.

chmod toma como parámetros los permisos en modo numérico y el nombre del archivo al que se desea realizar el cambio. Por ejemplo:

```
$ ls -la archivo_prueba
-rw-rw-r-- 1 edgar edgar 0 2008-09-19 04:21 archivo_prueba
$ chmod 755 archivo_prueba
$ ls -la archivo_prueba
-rwxr-xr-x 1 edgar edgar 0 2008-09-19 04:22 archivo_prueba
```

El comando `chmod` también se puede usar recursivamente para cambiar los permisos de todos los archivos dentro de una carpeta. Esto se hace con la opción `R`. Por ejemplo: “`chmod -R 755 nombre_carpeta/`”

Para cambiar el dueño y grupo asociado con un archivo determinado podemos usar el comando `chown`. Por ejemplo para cambiar los permisos del archivo anterior al usuario y grupo `root` ejecutamos lo siguiente.

```
# ls -la archivo_prueba
-rwxr-xr-x 1 edgar edgar 0 2008-09-19 04:23 archivo_prueba
# chown root.root archivo_prueba
# ls -la archivo_prueba
-rwxr-xr-x 1 root root 0 2008-09-19 04:24 archivo_prueba
```

Al igual que el comando `chmod`, el comando `chown` también soporta la opción `R`.

3.5 RPMs y actualizaciones vía Yum

Los paquetes de software que se incluyen en Elastix vienen en un formato llamado RPM. RPM es un formato heredado de Red Hat y su nombre significa Red Hat Package Manager por sus siglas en inglés. Esto quiere decir que los instaladores de muchos paquetes de software adicional pueden ser instalados en Elastix de manera sencilla.

Los RPMs también nos permitirán actualizar paquetes de software y se encargarán de resolver las dependencias necesarias para que dichos paquetes no causen problemas en el sistema. Esto se logra gracias a que el sistema RPM mantiene una base de datos interna de todos los paquetes instalados con sus respectivas dependencias y versiones. Además nos provee con un comando muy útil para manipular las instalaciones, desinstalaciones, actualizaciones y otras tareas importantes.

A continuación algunas opciones de uso del comando “`rpm`”:

| Acción a realizar | Comando |
|---------------------|--|
| Instalación de RPMs | <code>rpm -ivh nombre_del_paquete.rpm</code> |

| | |
|---|--|
| Actualización de un RPM | <code>rpm -Uvh nombre_del_paquete.rpm</code> |
| Eliminación de un RPM | <code>rpm -e nombre_del_paquete.rpm</code> |
| Obtener información de un archivo RPM | <code>rpm -qpi nombre_del_paquete.rpm</code> |
| Obtener un listado de todos los RPMs instalados en el sistema | <code>rpm -qa</code> |

Yum al rescate

La instalación manual de RPMs mediante el comando `rpm` está bien para muchas de las tareas, pero hay ocasiones en las cuales instalar RPMs de esta manera es tedioso y complicado. Un ejemplo de esto es cuando instalamos paquetes RPMs complejos que tienen muchas dependencias y al momento de instalar dicho paquete resulta que nos damos cuenta que tenemos que instalar otro paquete necesario y tenemos que buscarlo en Internet para instalarlo.

Muchas veces un RPM tiene decenas o hasta cientos de dependencias lo que por supuesto nos llevaría horas solucionar manualmente.

Adicionalmente a este problema de dependencias existe otro y es el de la confiabilidad de los RPMs. Es decir, luego de horas y horas de instalar nuestros paquetes manualmente nos damos cuenta de que uno de ellos sobrescribió un archivo importante del sistema y corrompió nuestro Elastix. Vaya dolor de cabeza!

Aquí es cuando entra en escena nuestro héroe llamado Yum!

Yum es una utilidad que nos ayuda a instalar paquetes RPMs desde repositorios y lo hace de una manera muy amigable.

La mayoría de las distribuciones serias, y Elastix no es la excepción, mantienen repositorios de paquetes oficiales de software accesibles gratuitamente a través de Internet y hacen el mayor esfuerzo porque dichos paquetes no tengan problemas de dependencias entre sí. Además los paquetes de los repositorios oficiales son probados para evitar molestias en los usuarios. Es por esta razón que si se usa Elastix lo recomendable es que se instalen paquetes desde el repositorio oficial.

Bueno, ahora vamos al grano y mostremos algunos comandos Yum útiles:

| Comando | Descripción |
|--|---|
| <code>yum update</code> | Actualizar TODO el sistema |
| <code>yum update nombre_paquete</code> | Actualiza solo el paquete especificado y todas sus dependencias |

| | |
|---|--|
| <code>yum install nombre_paquete</code> | Instala el paquete especificado y todas sus dependencias |
|---|--|

***Nota:** Cabe destacar que Elastix nos provee de una interfase Web amigable para actualizar paquetes y administrar repositorios. Por lo tanto, los procedimientos de actualización manuales aquí descritos se dejarán para casos excepcionales. Se recomienda en lo posible utilizar la interfase Web.*

4

Asterisk Essentials

Si una idea no es absurda al principio, entonces no merece la pena.

-- Albert Einstein

4.1 Qué es Asterisk?

Básicamente Asterisk es un software de central telefónica con capacidad para voz sobre IP que es distribuido bajo licencia libre.

Partiendo de este concepto muy básico Asterisk no es una central telefónica cualquiera; se trata de una central telefónica rica en características que en otros tiempos solo eran accesibles mediante la compra de productos costosos.

Esto ha hecho que muchas empresas consideren a Asterisk como una seria opción al momento de planificar su proyecto telefónico y por esta razón Asterisk ha tenido gran acogida a nivel mundial.

Al ver la oportunidad de negocio muchos fabricantes se han sumado a ofrecer hardware telefónico compatible con Asterisk, principalmente tarjetas PCI para conexión con la PSTN y esto ha hecho que la oferta de centrales telefónicas basadas en Asterisk crezca en los últimos tiempos.

Asterisk es uno de los componentes más importantes de Elastix y quien provee la mayoría de las características telefónicas de la distro.

4.2 Breve historia de Asterisk

Asterisk fue concebido y desarrollado por Mark Spencer. En un principio por una necesidad personal pues necesitaba una central telefónica para la pequeña empresa de soporte que estaba fundando llamada “*Linux Support Services*”.

Inicialmente pensó en adquirir una pero pronto se dio cuenta que estaba muy lejos de su presupuesto, así que luego de pensarlo un poco decidió crear la suya propia y comenzó a codificar lo que hoy conocemos como Asterisk. En 1999, cuando tuvo un código digno de mostrar al mundo decidió liberarlo bajo licencia GPL.

En cierto momento Mark se dio cuenta de que su software necesitaba interactuar con hardware telefónico y se topó con el proyecto Zaptel, un proyecto de código abierto creado por Jim Dixon, que tenía el objetivo de crear *drivers* abiertos para tarjetas telefónicas de computadora. A partir de allí Asterisk y Zaptel caminarían de la mano; tanto así que en la actualidad los dos proyectos son mantenidos por la misma compañía.

A pesar de lo que se pueda pensar, Mark no era un novato en el mundo del software libre. El ya había participado activamente en el desarrollo del cliente de mensajería Gaim, actualmente llamado Pidgin, uno de los más populares en el ámbito del software libre.

En el 2002 Linux Support Services se convertiría en Digium, quien actualmente ofrece productos y servicios relacionados con Asterisk y se encarga del desarrollo del producto.

4.3 Funcionalidad provista por Asterisk

Como ya hemos dicho Asterisk es un software de centralita rico en características. Al momento de escribir este libro las características soportadas más relevantes son las que enlistaremos a continuación.

- Contestación Automática de Llamadas
- Transferencia de Llamadas
- Opción de No Molestar
- Parqueo de Llamadas
- Contestación de una llamada a una extensión remota
- Monitoreo y Grabación de Llamadas

- Voicemail
- Conferencias
- Reportación de Llamadas
- Colas de atención
- Llamada en espera
- Identificador de Llamante
- Bloqueo por llamante identificado
- Recepción de Fax
- Listado Interactivo del directorio de extensiones
- Interactive Voice Response (IVR)
- Música en espera
- Manejo de comportamiento por tiempo (Time Conditions)
- Follow me

Contestación automática de llamada

Asterisk provee la facilidad de configurar una contestadora automática que ayuda al manejo del flujo de llamadas entrantes contestándolas automáticamente sin necesidad de una operadora en vivo. Diseñado para empresas que reciben altos flujos de llamadas o que no poseen personal necesario para manejar un alto tráfico de llamadas entrantes.

La contestadora responde a los dígitos marcados por el teclado de la persona que llama, enruta las llamadas a extensiones específicas, provee acceso a información pregrabada y toma mensajes 24 horas al día.

Transferencia de llamadas

Existen dos métodos para transferir llamadas: transferencias desatendidas y transferencias atendidas.

Transferencia atendida de llamadas

Este método es el usual, es decir, la llamada que usted está atendiendo, es transferida a una extensión, en donde primero la extensión a la cual es transferida contesta, usted hace la presentación de la llamada y cuelga su extensión, en este caso la llamada queda conectada a la nueva extensión.

Si la nueva extensión no desea que se le transfiera la extensión, simplemente deberá colgar su teléfono, en cuyo caso, la llamada será nuevamente conectada a la extensión

original. Mientras el proceso de transferencia se completa, el llamante de la llamada externa escuchará la música “Music On Hold”.

Para realizar una transferencia atendida, usted debe digitar un código predeterminado. En ese momento un mensaje pedirá que le digite la extensión donde desea transferir la llamada, una vez discada dicha extensión, usted escuchará el ringeado de la llamada hacia la nueva extensión y podrá establecer una comunicación preliminar con esta extensión. Cuelgue su teléfono y la llamada externa quedará conectada a la nueva extensión.

Transferencia desatendida de llamadas

Este método permite transferir una llamada sin establecer una comunicación previa con la extensión a la cual se desea transferir la llamada.

Para transferir una llamada de manera no atendida, mientras atiende una llamada digite en el teléfono un código predeterminado. Usted escuchará un mensaje pidiendo el número de la extensión a la cual desea transferir la llamada y una vez discada usted recibirá el tono de ocupado y la llamada ha quedado conectada a la nueva extensión.

Opción de No Molestar

Esta opción permite a cualquier usuario configurar su extensión para que no reciba llamadas por un período que el crea conveniente. Cualquier llamada entrante a esta extensión será ruteada automáticamente al buzón de correos de la misma.

Para habilitar la opción de No Molestar y para deshabilitar esta opción, en el teclado de su teléfono digite los respectivos códigos predeterminados.

Parqueo de llamadas

El parqueo de llamadas permite al usuario que recibe una llamada, enviar su llamada a un “cuarto” de parqueo, para volver a atenderla desde otra extensión. Este tipo de acción es útil en el siguiente ejemplo. Supongamos que el administrador de la red recibe una llamada a su extensión ubicada en su escritorio, y esta requiere que él se ubique de frente en un servidor que se encuentra tres pisos mas arriba, y no sabe si alguien podrá atender la llamada en el teléfono ubicado en el cuarto de equipos para que pueda hacer la transferencia normal de la llamada. En este caso, la opción es enviar la llamada a un cuarto temporal, donde mientras tanto el llamante escuchará la música en espera configurada, hasta que el administrador llegue al cuarto de equipos y vuelva a tomar la llamada.

En su central telefónica, para enviar la llamada actual para un cuarto de parqueo, transfiera la llamada a una extensión predefinida, utilizando el carácter predeterminado

para transferir. Una vez digitada la extensión, usted escuchará el número de extensión que deberá marcar desde el otro teléfono para volver a recuperar la llamada. La llamada quedará en espera por un lapso máximo de 2 minutos. Mientras la llamada está parqueada, el que está llamando escuchará la música configurada como “*Music On Hold*”.

Contestación de una llamada a una extensión remota

Esta característica permite atrapar una llamada que se encuentra timbrando en una extensión que no es la suya de manera remota. Para atrapar la llamada timbrando en una extensión, digite en su teléfono el predeterminado para esta función.

Monitoreo y Grabación de Llamadas

Esta característica de Asterisk permite dar seguimiento a las llamadas, por ejemplo para fines de control de calidad del desenvolvimiento de los operadores telefónicos o de los agentes de ventas. Para esto existe la facilidad de que a través de la digitación de un código predeterminado se ejecute un comando que permita escuchar en línea la conversación sostenida desde cualquier de las extensiones.

Adicionalmente existe la facilidad de grabar las conversaciones de cualquier extensión en forma aleatoria ó programada previamente. Estas grabaciones se almacenarán en el disco duro del servidor Asterisk para su posterior revisión.

Buzón de Mensajes de Voz (*VoiceMail*)

El buzón de mensajes de voz es una aplicación que permite escuchar mensajes dejados por llamadas que no pudieron ser atendidas por la extensión.

Para acceder al buzón de mensajes, digite en su teléfono el código predeterminado respectivo. Se le pedirá que ingrese el número de la extensión y la clave que debe haber sido suministrada por el administrador del sistema. Una vez ingresados estos dos datos, una grabación le indicará si tienes nuevos mensajes de voz, y además opciones para grabar mensajes personalizados para ser usados en esta aplicación.

Conferencias

Por defecto, cada extensión tiene asociado un cuarto de conferencias. Este cuarto de conferencias puede ser utilizado por cualquier usuario que pida a sus compañeros que ingresen a su cuarto y así, poder establecer una comunicación multiusuarios.

También es posible enviar una llamada externa a un cuarto de conferencias, simplemente transfiriendo la llamada al cuarto de conferencias.

Los cuartos de conferencias están asociados a los números de extensiones, anteponiendo un número predeterminado, Por ejemplo:

Si su extensión es 101 y el número para esta función es 8, usted tiene asociado el cuarto de conferencias 8101, por lo que si usted desea establecer una llamada multiusuarios, en su teléfono digite la extensión “8101”, e ingresará al cuarto de conferencias; pida a las personas con las cuales desea establecer la conferencia que en sus teléfonos también digiten “8101” y podrá conversar con cualquier usuario que ingrese a ese cuarto. Al primer usuario que ingresa a un cuarto de conferencias, una grabadora le indicará que es el único usuario en el cuarto de conferencias.

Si desea establecer una conferencia incluyendo una llamada externa, en el momento que esté atendiendo dicha llamada, haga una transferencia desatendida a la extensión 8101; al usuario que usted envió al cuarto de conferencias se le indicará mediante un mensaje que es el único usuario en el salón de conferencias. Ahora usted debe discar en su teléfono el número del salón de conferencias e indicar al resto de participantes que ingresen al mismo salón.

Reportación de número marcados

Asterisk genera CDRs (*Call Detail Records*) o Registros de Detalle de Llamadas y los puede almacenar en una base de datos. Accediendo a esta base de datos, se pueden generar reportes que detallan qué extensión llamó a qué número, si la llamada fue contestada o no, cuánto duró la llamada, por qué puerto o línea se realizó la llamada telefónica.

Colas de Atención

Esta característica permite que un ilimitado número de llamantes puedan permanecer en espera hasta que un representante o recurso esté disponible para dar asistencia. Esto permite que usted provea a sus clientes la misma calidad de servicio. Adicionalmente, esto asegura que sus clientes no terminarán obligatoriamente en el *voicemail*, lo cual puede resultar molesto para el llamante. Es decir, con esto nos aseguramos que el llamante siempre tenga la oportunidad de ser atendido por una persona. El tiempo que el llamante permanece en espera puede aprovecharse para darle a conocer

más servicios o productos de su compañía, ofertas especiales, anuncios informativos ó se puede proporcionar música en espera.

Llamada en espera

Esta característica permite que la persona que se encuentra atendiendo una llamada y recibe otra pueda interrumpir temporalmente su primera conversación para atender la segunda llamada y poder acordar un tiempo para devolver o atender su llamada. Debemos recordar que esta característica afecta muchas veces a las conexiones telefónicas para transferencia de datos, por lo que es común que los clientes la deshabiliten.

Identificador de llamante

Esta señal es enviada entre las señales de RING o durante el proceso de establecimiento de la llamada, antes de que sea contestada. Asterisk aprovecha esta facilidad y a nivel extensiones IP soporta plenamente su manejo. Sin embargo a nivel de líneas de la red telefónica pública conmutada es el proveedor de estas quien debe habilitar o proporcionar esta característica. Se la conoce también como Caller Display ó *Calling Line Identification Presentation*.

Bloqueo por llamante identificado

Esta característica previene que alguien con identificador de llamante vea el número desde el que usted llama. Esta facilidad de ocultar el ID puede ser completa o selectiva. No todas las redes de telefonía pública soportan esta característica.

En Asterisk esta característica también describe el bloquear una llamada entrante en función del ID.

Envío y recepción de Fax

Asterisk permite detectar automáticamente cuando un llamante está intentando enviar un fax. Se puede poner como parte del mensaje de bienvenida de la central una frase que diga: “Si quiere enviar un fax, envíelo ahora”. Esto nos ayuda a eliminar papel, prescindir de máquinas ó puertos telefónicos exclusivos para fax y digitalizar la información recibida como fax. Luego de ser digitalizado, este documento puede ser enviado vía e-mail a una cuenta específica para su revisión.

Así también Asterisk puede ser configurado como Fax Server para que los documentos que son enviados a una cuenta de correo específica automáticamente sean enviados como faxes.

Listado interactivo del directorio de extensiones

Asterisk puede contener en su base de datos el directorio telefónico del personal de la empresa (nombre, apellido y extensión). Esta facilidad permite por ejemplo, que la

persona que llama pueda digitar desde su teléfono los números correspondientes a las 4 primeras letras del apellido ó nombre de la persona con la que desea hablar. Esto puede proporcionar una alternativa para llamantes que no conocen la extensión de la persona que quieren contactar pero conocen solo su apellido ó nombre.

Interactive Voice Response (IVR)

A través de esta característica se proporciona acceso a opciones telefónicas que mejorará la forma en que un sistema telefónico acepta y distribuye sus llamadas. Con un menú de IVR usted podrá distribuir sus llamadas de una manera precisa que ayudará a incrementar el desempeño de sus empleados y la satisfacción de sus clientes, ya que se agiliza el acceso a los servicios y a la información que posee su compañía. Con esta facilidad se proporcionan procesos de auto atención que permitirán incluso prescindir de una recepcionista.

Con esta característica se pueden crear menús de activación por dígitos ó por comandos de voz para ejecutar incluso transacciones a través de tarjetas de créditos ó consultas a base de datos.

4.4 Funcionamiento de Asterisk

Encendido y apagado

A pesar de que Asterisk puede ser controlado directamente con su *script* de inicio (ubicado en /etc/rc.d/init.d/asterisk) lo recomendable en Elastix es hacerlo mediante el programa “amportal”. Esto es así debido a que existen otros programas relacionados con Asterisk que también se deben sincronizar con el encendido y apagado de Asterisk. Para apagarlo podemos ejecutar:

```
# amportal stop
```

Para encenderlo:

```
# amportal start
```

Para reiniciarlo:

```
# amportal restart
```

Directorios de Asterisk

Asterisk organiza sus archivos en algunos directorios. Entre los más importantes tenemos a los siguientes.

| Directorio | Descripción |
|----------------------------|---|
| /etc/asterisk/ | Aquí residen los archivos de configuración de asterisk |
| /usr/lib/asterisk/modules/ | Este directorio contiene los módulos de Asterisk |
| /usr/sbin/ | Aquí reside el binario de Asterisk |
| /var/log/asterisk/ | Contiene los logs de Asterisk |
| /var/lib/asterisk/agi-bin/ | Directorio donde residen los <i>scripts</i> AGI |
| /var/lib/asterisk/mohmp3 | Carpeta que contiene archivos para música en espera |
| /var/lib/asterisk/sounds | Sonidos que Asterisk utiliza como <i>prompts</i> de voz |
| /var/spool/asterisk/ | Directorio donde Asterisk guarda archivos que genera producto de su funcionamiento como <i>voicemails</i> y grabaciones de llamadas |
| /var/run/ | Archivos con información de PIDs |
| /var/log/asterisk/ | Aquí residen los archivos de <i>log</i> de Asterisk como el <code>/var/log/asterisk/full</code> o el <i>log</i> de texto de CDRs |

Estructura modular

Asterisk utiliza la idea de módulos para extender su funcionalidad. Es tan importante la funcionalidad residente en los módulos que sin ellos Asterisk por sí solo no cumpliría ninguna función interesante.

Estos módulos residen en la carpeta `/usr/lib/asterisk/modules/` y son archivos con extensión `.so`

Los módulos pueden ser controlados a través del archivo de configuración `modules.conf`. En este archivo puedo decirle a Asterisk que cargue o no cargue un módulo específico a través de las directivas `load` y `no load`. Por omisión en Elastix todos los módulos tratan de ser cargados.

A continuación un fragmento del archivo `modules.conf` que viene con Elastix, donde se puede observar cómo se le dice a Asterisk que no cargue algunos módulos en el arranque.

```
;
; DON'T load the chan_modem.so, as they are obsolete in * 1.2

noload => chan_modem.so
noload => chan_modem_aopen.so
noload => chan_modem_bestdata.so
noload => chan_modem_i4l.so

; Trunkisavail is a broken module supplied by Trixbox
noload => app_trunkisavail.so

; Ensure that format_* modules are loaded before res_musiconhold
;load => format_ogg_vorbis.so
load => format_wav.so
load => format_pcm.so
load => format_au.so
```

Adicionalmente se pueden cargar módulos en tiempo real a través de la consola de Asterisk con el comando “`module load`”. Veremos más de la consola de Asterisk o CLI más adelante.

4.5 El proyecto Zaptel, ahora DAHDI

A pesar de ser un gran producto de software Asterisk no puede hacer todo el trabajo. Uno de los paquetes de software que lo complementa se llama Zaptel.

Básicamente Zaptel es un conjunto de *drivers* para controlar hardware telefónico como tarjetas PCI que nos permiten conectarnos a la PSTN.

Estos *drivers* se comunican con Asterisk a través de un módulo de Asterisk llamado `chan_zap.so` que se configura a través del archivo `zapata.conf`.

Adicionalmente Zaptel tiene su propio archivo de configuración, independiente de Asterisk, llamado `zaptel.conf` y ubicado en la carpeta `/etc/`.

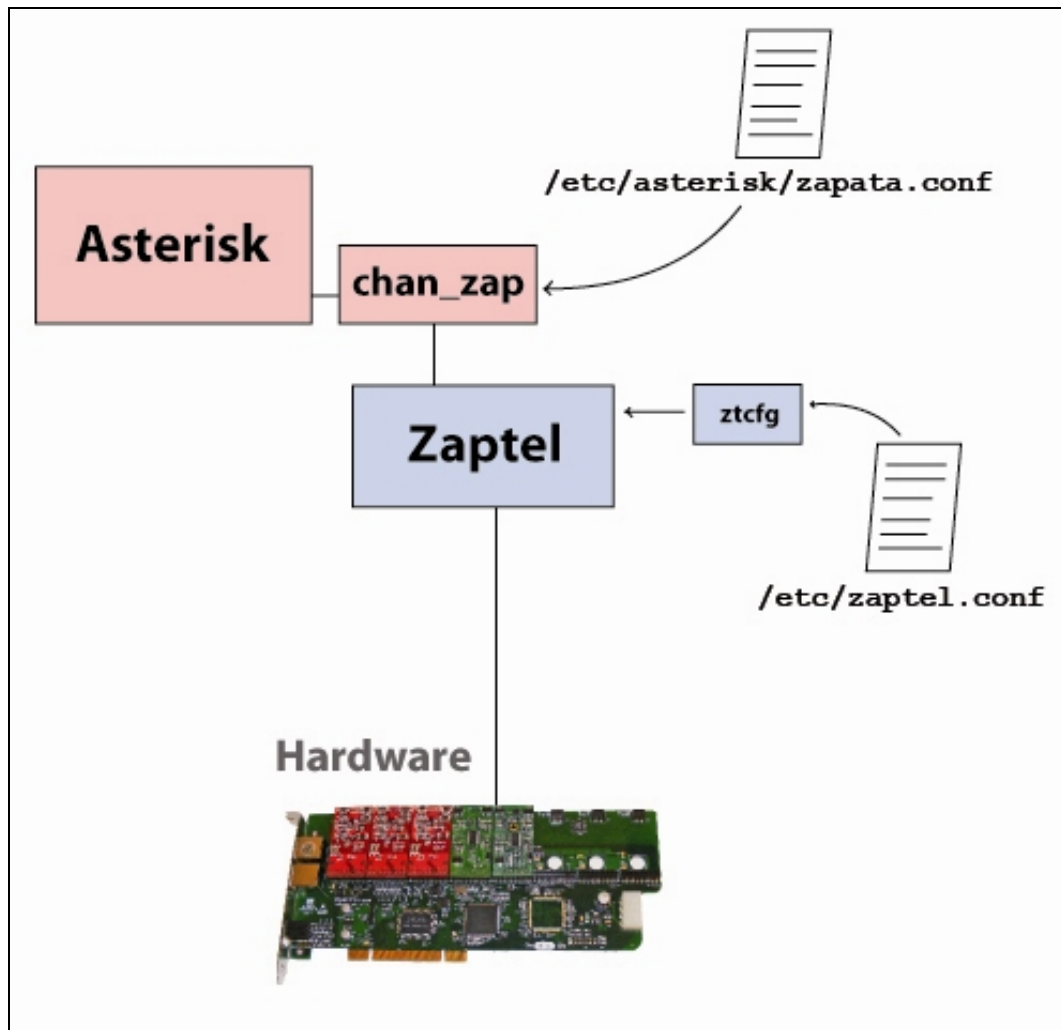


Diagrama explicativo de la interacción de Asterisk con Zaptel

Breve historia del proyecto Zaptel

En el año 2000 Jim Dixon lanzó el proyecto Zaptel cuyo nombre era una abreviación de **Zapata Telephony Project**. Lo nombró así en honor a Emiliano Zapata, héroe de la revolución mexicana.

Los drivers fueron lanzados bajo licencia GPL de tal manera que cualquiera pudo tener acceso al código.

Inicialmente Zaptel contenía *drivers* para tarjetas Tormenta fabricadas por Zapata Telecom, pero pronto Digium comenzó a mejorar los *drivers* y extender el soporte para nuevos modelos de hardware, entre ellos los que el mismo Digium fabricaba. Pronto Digium se convirtió en el principal desarrollador de Zaptel.

Luego de algunos años de mantener Zaptel, Digium cayó en cuenta de que Zaptel era una marca registrada de Zapata Telecom y para evitar cualquier posible confusión futura decidió cambiar de nombre a sus *drivers*. El nombre elegido es DAHDI.

DAHDI ya se encuentra disponible para descarga pero muchos continúan usando Zaptel por razones históricas o esperando a que este nuevo paquete se estabilice un poco más.

Digium ha manifestado que en su nueva versión de Asterisk 1.6 ya no soportarán Zaptel oficialmente.

4.6 Configuración de Asterisk

Archivos de configuración

Asterisk se puede configurar a través de algunos archivos de configuración ubicados en la ruta `/etc/asterisk`. Existen decenas de archivos de configuración en este directorio y se encuentran en texto plano para facilitar su modificación por lo que nosotros podremos modificarlos desde la línea de comandos utilizando el editor `vim` o el editor de línea de comandos de nuestra preferencia.

Si bien todos estos archivos son importantes no todos son necesarios y existen otros que ya han sido pre-configurados por Elastix para que no tengamos que modificarlos. Algunos de los más importantes se explican en la siguiente tabla.

| Archivo | Descripción |
|------------------------------|---|
| <code>extensions.conf</code> | Aquí reside el plan de marcado. En Elastix este archivo incluye otros más para organizar el plan de marcado de mejor manera. Estos archivos adicionales empiezan con la cadena <code>extensions_</code> |
| <code>sip.conf</code> | Aquí se definen los endpoints SIP |
| <code>iax.conf</code> | Aquí se definen los endpoints IAX |
| <code>zapata.conf</code> | Archivo de configuración de los canales tipo ZAP. Aquí se puede troncalizar dichos canales y configurar algunos parámetros |

Comentarios en los archivos de configuración

Los archivos de configuración de Asterisk pueden contener comentarios. Un comentario empieza con el carácter de punto y coma, y una vez escrito el resto de la línea se convierte en un comentario. Es decir que es ignorado por Asterisk al interpretar el archivo. Veamos un ejemplo de comentario.

```
;
; El siguiente es un bloque de comentarios.
; Como pueden ver puedo escribir cualquier cosa
; aquí porque será ignorada por Asterisk.
; Los comentarios son muy útiles para escribir
; notas e información relativa a la configuración
; que estamos escribiendo.

; Ahora si continúo con las directivas...
;

#include sip_general_custom.conf
#include sip_nat.conf
#include sip_registrations_custom.conf
#include sip_registrations.conf
```

4.7 Plan de marcado (dial plan)

El plan de marcado en Elastix reside principalmente en el archivo `extensions.conf`. Desde este archivo se incluyen otros importantes que contienen también porciones del plan de marcado. Los principales son:

- `extensions_additional.conf`: Aquí reside el plan de marcado variable escrito por freePBX. Si se escribe manualmente en este archivo los cambios se perderán cuando freePBX sobrescriba este archivo.
- `extensions_custom.conf`: Aquí se puede escribir plan de marcado definido manualmente por el usuario. Este archivo no es tocado por el freePBX por lo que es seguro escribirlo sin temor a perder los cambios.

Todos estos archivos de plan de marcado están escritos en texto plano en un formato definido por Asterisk. Estos archivos se encuentran divididos en secciones llamadas contextos.

Contextos

Básicamente los contextos sirven para agrupar lógica de plan de marcado. Los contextos también pueden incluir otros contextos mediante la cláusula "*include*" o redirigir el hilo de ejecución del plan de marcado a otros contextos dependiendo de condiciones. Por ejemplo, un contexto puede contener la lógica necesaria para rutear una llamada hacia la troncal correspondiente luego de examinar el prefijo del número telefónico marcado. Un contexto también puede agrupar la lógica necesaria para la aplicación de reloj despertador. También puede incluir lógica para acceder al *voicemail* o para autenticar un número de PIN. En general prácticamente todo en el plan de marcado se encuentra agrupado en contextos. Los contextos son los agrupadores u organizadores del plan de marcado.

Cada contexto debe llevar un nombre único y este nombre va encerrado entre los símbolos [y]. A continuación un ejemplo donde se pueden observar dos contextos llamados `app-pickup` y `ext-test`

```
[app-pickup]
include => app-pickup-custom
exten => _*.*,1,Noop(Attempt to Pickup ${EXTEN:2})
exten => _*.*,n,Pickup(${EXTEN:2})

[ext-test]
include => ext-test-custom
exten => 7777,1,Goto(from-pstn,s,1)
exten => 666,1,Goto(ext-fax,in_fax,1)
exten => h,1,Macro(hangupcall,)
```

Fragmento del archivo `extensions_additional.conf` para ilustrar los contextos

Contextos reservados

Hay dos nombres de contextos que están reservados para un propósito especial y son [general] y [globals] .

Instrucciones

Los contextos contienen instrucciones. Estas instrucciones son líneas de plan de marcado que tienen la siguiente sintaxis.

```
exten => nombre_extension, prioridad, aplicación
```

Estas instrucciones también son llamadas extensiones.

***Nota:** Normalmente asociamos el término extensión a un número de teléfono, pero en Asterisk el término extensión va más allá de lo que normalmente conocemos. Una extensión es una serie de lógica de plan de marcado. Es más, cuando marcamos un número desde nuestro teléfono no necesariamente tiene que ringar otro teléfono pues podemos crear la lógica que queramos para este número dado. Por ejemplo podemos hacer que la llamada simplemente se cuelgue, que reproduzca un mensaje dado o cosas mucho más complejas.*

Una típica instrucción luce como la siguiente.

```
exten => 4567,1,Answer()
```

Nombre de Extensión

El nombre de extensión es un identificador numérico de la extensión. Llamémosle el número de teléfono por ahora para simplificar las cosas. Sin embargo hay una extensión especial no numérica llamada extensión **s** que le prestaremos particular atención.

La extensión “s” significa cualquier extensión. La letra s viene de la palabra “start” que significa inicio. Un ejemplo del uso de la extensión s es el siguiente.

```
[prueba]
exten => s,1,Answer()
exten => s,2,noOp("Prueba")
```

La extensión **s** se usa cuando el usuario que realizo la llamada aun no ha presionado el numero de la extensión. Por ejemplo una llamada entrante que todavía se encuentra escuchando el IVR de bienvenida.

Prioridad

La prioridad indica el orden en el que debe ser ejecutada una instrucción dentro de un contexto. Es necesario que la primera prioridad sea la 1 pues cuando Asterisk lee un contexto lo primero que busca es esta prioridad.

Existe también una prioridad especial que es la prioridad **n**. La prioridad n significa que a la prioridad de la instrucción anterior se le debe sumar uno. Esto es muy útil para facilitar la escritura del plan de marcado.

Por ejemplo, el siguiente plan de marcado:

```
exten => s,1,NoOp("Hola este es solo un ejemplo")
exten => s,2,NoOp("para demostrar")
exten => s,3,NoOp("como usar la prioridad n")
```

Podría ser escrito de la siguiente manera usando la prioridad n.

```
exten => s,1,NoOp("Hola este es solo un ejemplo")
exten => s,n,NoOp("para demostrar")
exten => s,n,NoOp("como usar la prioridad n")
```

Las prioridades también pueden contener etiquetas para facilitar su identificación ya que si usamos siempre la prioridad “n” será difícil acceder a una ubicación determinada dentro de un contexto. Esto se consigue con la sintaxis `n(nombre_etiqueta)`.

Aplicación

En las aplicaciones radica el verdadero poder del plan de marcado en Asterisk. Para hacer una analogía, las aplicaciones son el equivalente a las funciones de un lenguaje de programación. Las aplicaciones nos permitirán contestar una llamada o colgarla, reproducir música en espera, saltar a otros contextos entre otras muchas cosas.

Las aplicaciones también pueden recibir parámetros. Por ejemplo a la aplicación *Dial()* habrá que indicarle qué número marcar para que pueda realizar su labor.

Más adelante en el presente capítulo se explicarán con más detalle las aplicaciones más comunes en el plan de marcado.

Variables

Las variables en el plan de marcado son un concepto similar a las variables en un lenguaje de programación. Es decir que son abstracciones que pueden almacenar información de naturaleza variable. Una variable tiene la siguiente sintaxis.

`${NOMBRE}`

Donde nombre es el nombre de la variable. Las variables pueden ser de tres tipos:

Globales

Son las variables definidas en la sección [globals] del `extensions.conf`. La palabra “global” quiere decir que pueden ser referenciadas desde cualquier lugar.

De canal

Son las variables que se pueden definir usando el comando (o aplicación) "Set".

Pre-definidas

Asterisk se encarga de administrar algunos nombres de variables por su cuenta. Estos nombres se encuentran predefinidos y se llenan con cierta información dependiendo del comportamiento de la llamada o de otros factores. Algunas variables predefinidas son:

```
${CALLERID}
${CALLERIDNAME}
${CALLERIDNUM}
${CHANNEL}
${CONTEXT}
${EPOCH}
${EXTEN}
${SIPUSERAGENT}
${UNIQUEID}
```

***Nota:** Hay un tipo adicional de variable que será introducida en Asterisk 1.6 llamada "Shared" pero no la estudiaremos aquí puesto que la última versión estable de Elastix disponible usa Asterisk 1.4*

A continuación un segmento del archivo `extensions.conf` que se distribuye con Elastix. Podemos ver aquí algunas variables predefinidas como `${UNIQUEID}`, `${BLINDTRANSFER}` y `${EPOCH}`, y variables de canal como `${CALLFILENAME}`.

```
[macro-record-enable]
exten => s,1,GotoIf($[${LEN}(${BLINDTRANSFER})] > 0]?2:4)
exten => s,2,ResetCDR(w)
exten => s,3,StopMonitor()
exten => s,4,AGI(recordingcheck,${STRFTIME}(${EPOCH},,%Y%m%d-%H%M%S)},${UNIQUEID})
exten => s,5,Noop(No recording needed)
exten => s,999,MixMonitor(${CALLFILENAME}.wav)
```

Aplicaciones más comunes

Answer

Sintaxis: Answer([delay])

Contesta un canal si este está timbrando. Puede recibir opcionalmente el parámetro `delay` que le indica si debe esperar un número determinado de milisegundos antes de contestar.

Background

Sintaxis:

```
Background(filename1[&filename2...][|opciones[|langoverride
][|context]])
```

Reproduce un listado de archivos de audio en el fondo, es decir que devuelve el control a Asterisk, quien puede continuar ejecutando el plan de marcado mientras el audio continúa siendo reproducido. Cuando termina de reproducir el último archivo termina su ejecución por lo que si se quiere seguir esperando por una extensión se debe usar otra aplicación en conjunto llamada `WaitExten`.

Esta aplicación es típicamente usada para reproducir el mensaje de bienvenida de un IVR.

La aplicación Background puede recibir algunas **opciones**:

| Opción | Descripción |
|--------|---|
| s | Saltarse la reproducción del audio si el canal no se encuentra contestado |
| n | No contestar el canal antes de reproducir los archivos |
| m | Dejar de reproducir el audio si un dígito marcado coincide con una |

| | |
|--|--|
| | extensión existente en el contexto destino |
|--|--|

Playback

Sintaxis: `Playback(filename1 [&filename2...] [, options])`

Reproduce uno o más archivos de audio. La diferencia con la aplicación Background es que Playback reproduce todo el archivo de audio hasta el final y no retorna el control hasta que termina la reproducción.

***Nota:** No es necesario especificar la extensión del archivo de audio en esta función. Los archivos de audio son buscados en el directorio /var/lib/asterisk/sounds*

Hangup

Sintaxis: `Hangup()`

Cuelga el canal y retorna -1

Goto

Sintaxis: `Goto([context|]extension|]priority)`

Salta la ejecución del plan de marcado a un contexto, extensión y prioridades dados. Si solo se pasa un parámetro se sobreentiende que se trata de una prioridad dentro del mismo contexto.

Dial

Sintaxis: `Dial(type/identifier, timeout, options, URL)`

Este comando es el que hace la magia. Marca un canal especificado y lo vincula con el canal corriente.

4.8 Asterisk CLI

Asterisk CLI es el nombre que recibe la consola de Asterisk. Es decir, una línea de comandos para controlar Asterisk directamente.

Para ingresar al CLI debemos ejecutar el siguiente comando desde la consola de Linux.

```
# asterisk -r
Asterisk 1.4.21.2, Copyright (C) 1999 - 2008 Digium, Inc. and others.
Created by Mark Spencer <markster@digium.com>
Asterisk comes with ABSOLUTELY NO WARRANTY; type 'core show warranty' for
details.
This is free software, with components licensed under the GNU General Public
License version 2 and other licenses; you are welcome to redistribute it under
certain conditions. Type 'core show license' for details.
=====
Connected to Asterisk 1.4.21.2 currently running on elastix (pid = 2245)
Verbosity is at least 3
elastix*CLI>
```

Como podemos observar obtenemos un prompt **CLI>** desde donde podemos ejecutar una serie de comandos que veremos en breve.

El CLI también nos puede proveer información en tiempo real de la actividad de Asterisk. Podemos controlar el grado de detalle con el que queremos ver dicha información con algunos comandos. Los más usados son los siguientes.

```
elastix*CLI> core set debug 9
Core debug was 3 and is now 9
elastix*CLI> core set verbose 9
Verbosity is at least 9
```

Mientras más altos los niveles de las variables debug y verbose más detallada será la información mostrada. Por omisión el grado nivel de verbose en Elastix se encuentra en 3.

***Nota:** Debido al gran volumen de información que se puede generar con estos comandos es importante reestablecer los valores de las variables debug y verbose a los mínimos al finalizar nuestras tareas de diagnóstico. Caso contrario Asterisk continuará generando esta información detallada cuando realmente no la necesitamos, lo cual es un trabajo extra que consume recursos. En sistemas con mucha actividad este gasto extra puede ser significativo y afectar el desempeño de las llamadas.*

Comandos del CLI

A continuación una breve lista de los comandos de CLI más comunes. Para ver una lista completa de comandos revisar el Apéndice C.

| Comando | Descripción |
|--------------------------------------|---|
| <code>agi debug</code> | Permite habilitar el <i>debug</i> de <i>scripts</i> AGI. Se apaga con “ <code>agi debug off</code> ” |
| <code>core set debug channel</code> | Habilita el <i>debug</i> en un canal determinado |
| <code>core set debug</code> | Permite cambiar el nivel de <i>debugging</i> . Se apaga con “ <code>core set debug off</code> ” |
| <code>core set verbose</code> | Incrementa la intensidad del <i>logging</i> . Muy útil en diagnóstico |
| <code>core show channels</code> | Muestra información de los canales activos en el instante de ejecutar el comando |
| <code>dialplan reload</code> | Recarga todo el plan de marcado |
| <code>dialplan show</code> | Muestra el plan de marcado |
| <code>help</code> | Muestra un listado con todos los comandos de CLI disponibles |
| <code>iax2 set debug</code> | Habilita el <i>debug</i> a nivel de IAX2. Se deshabilita con el comando “ <code>iax2 set debug off</code> ” |
| <code>iax2 show peers</code> | Muestra los peers definidos y alguna información adicional como el IP desde donde se conectan, el estatus, entre otros datos |
| <code>iax2 show registry</code> | Muestra información de registro IAX2 e información como el estatus y el IP a la que se trata de conectar |
| <code>module reload</code> | Recarga todos los módulos de Asterisk. Útil para aplicar los cambios hechos en los archivos de configuración |
| <code>pri debug span</code> | Habilita <i>debug</i> de un <i>span</i> determinado de una interfase PRI |
| <code>restart gracefully</code> | Reinicia Asterisk de manera ordenada |
| <code>restart now</code> | Reinicia Asterisk de manera brusca |
| <code>restart when convenient</code> | Reinicia Asterisk cuando no haya actividad en la central. Adecuado en ambientes de producción donde no tenemos otra opción que reiniciar la central pero no queremos tampoco que se corten llamadas |

| | |
|-----------------------------------|--|
| <code>sip set debug ip</code> | Habilita <i>debug</i> de SIP a nivel de una dirección IP específica |
| <code>sip set debug peer</code> | Habilita debug de SIP a nivel de un <i>peer</i> específico |
| <code>sip show peers</code> | Muestra un listado con todos los <i>peers</i> SIP configurados e información de cada uno de ellos como el estatus |
| <code>sip show registry</code> | Muestra un listado con los registros SIP e información relevante de cada uno de ellos |
| <code>stop gracefully</code> | Detiene Asterisk de manera ordenada |
| <code>stop now</code> | Detiene Asterisk de manera brusca |
| <code>stop when convenient</code> | Detiene Asterisk cuando no haya actividad en la central. Adecuado en ambientes de producción donde no tenemos otra opción que reiniciar la central pero no queremos tampoco que se corten llamadas |
| <code>zap show channels</code> | Muestra un listado de los canales ZAP y algunos parámetros de funcionamiento. Ver ejemplo luego de esta tabla. |
| <code>zap show status</code> | Muestra un listado de los dispositivos ZAP y un reporte de alarmas y errores |

| elastix*CLI> zap show channels | | | | |
|--|-----------|-------------|----------|---------------|
| Chan | Extension | Context | Language | MOH Interpret |
| pseudo | | from-zaptel | es | default |
| 1 | | from-pstn | | default |
| 2 | | from-pstn | | default |
| 3 | | from-pstn | | default |
| 4 | | from-pstn | | default |
| 5 | | from-zaptel | es | default |
| 6 | | from-zaptel | es | default |
| 7 | | from-zaptel | es | default |
| 8 | | from-pstn | | default |
| 9 | | from-zaptel | es | default |
| 10 | | from-zaptel | es | default |
| 11 | | from-zaptel | es | default |
| 12 | | from-zaptel | es | default |
| elastix*CLI> | | | | |

5

Instalando Elastix

Hay hombres que luchan un día y son buenos. Hay otros que luchan un año y son mejores. Hay quienes luchan muchos años, y son muy buenos. Pero hay los que luchan toda la vida, esos son los imprescindibles

-- Bertolt Brecht

5.1 Instalando desde CD

Antes de comenzar asegúrese de tener una copia de la última versión estable de Elastix. Esta se puede conseguir en el sitio de descargas del proyecto en <http://sourceforge.net/projects/elastix/>

Elastix se distribuye como un archivo ISO que puede ser quemado a un CD desde cualquier software de grabación de CDs.

Una vez quemado el CD insértelo en su computador al momento de encenderlo. Asegúrese de que su computador arranque de la unidad de CDROM, caso contrario deberá habilitar esta opción en el BIOS de su máquina. Si todo va bien debería obtener una pantalla como la siguiente.



```
- To install or upgrade in graphical mode, press the <ENTER> key.  
- To install or upgrade in text mode, type: linux text <ENTER>.  
- Use the function keys listed below for more information.  
[F1-Main] [F2-Options] [F3-General] [F4-Kernel] [F5-Rescue]  
boot: _
```

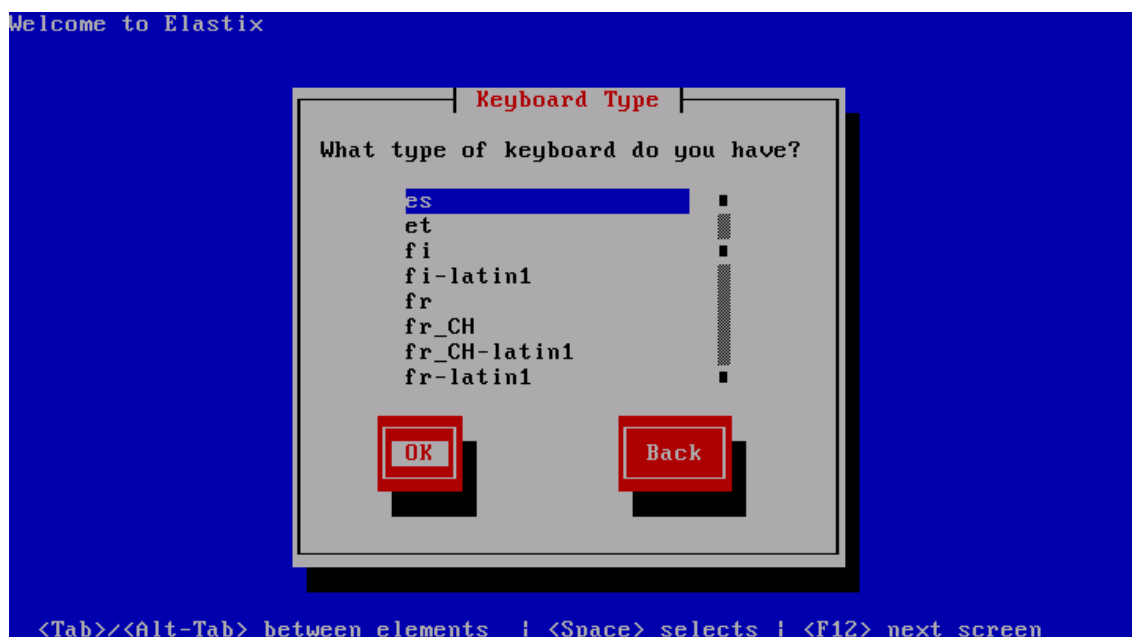
Pantalla de instalación inicial

***Nota:** El CD de instalación de Elastix formateará TODO el disco duro durante el proceso de instalación así que asegúrese de no tener información que vaya a necesitar en su disco duro.*

Si usted es un usuario experto puede ingresar en modo avanzado digitando el comando:
advanced

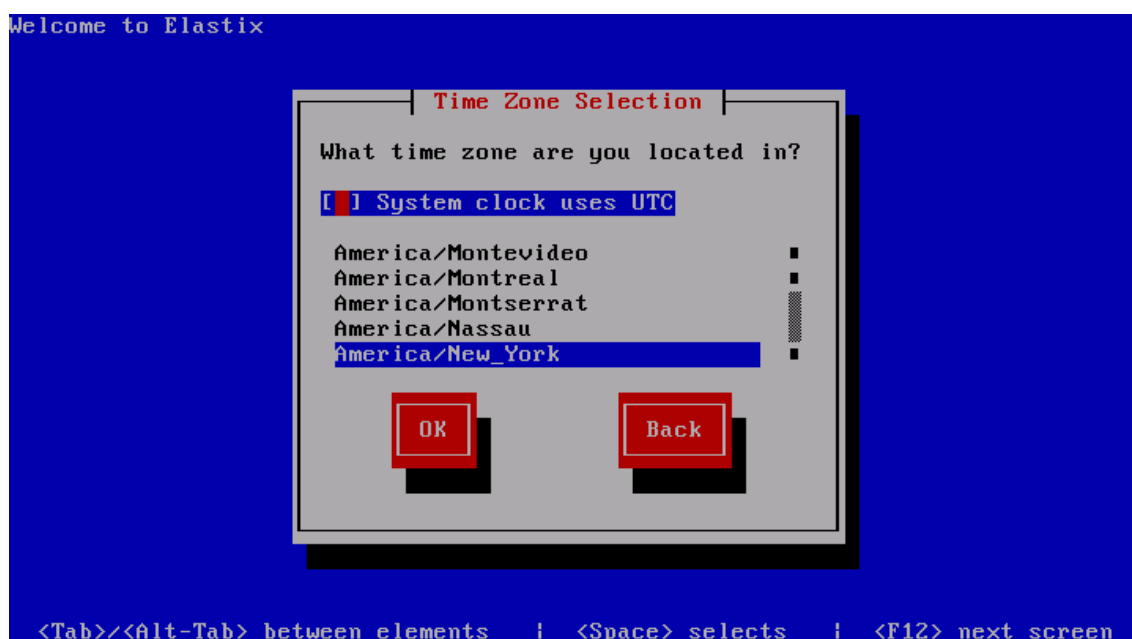
Caso contrario espere, el CD de instalación iniciará la instalación automáticamente ó presione ENTER.

Proceda a escoger el tipo de teclado de acuerdo al idioma. Si su teclado es de idioma español seleccione la opción “es”.



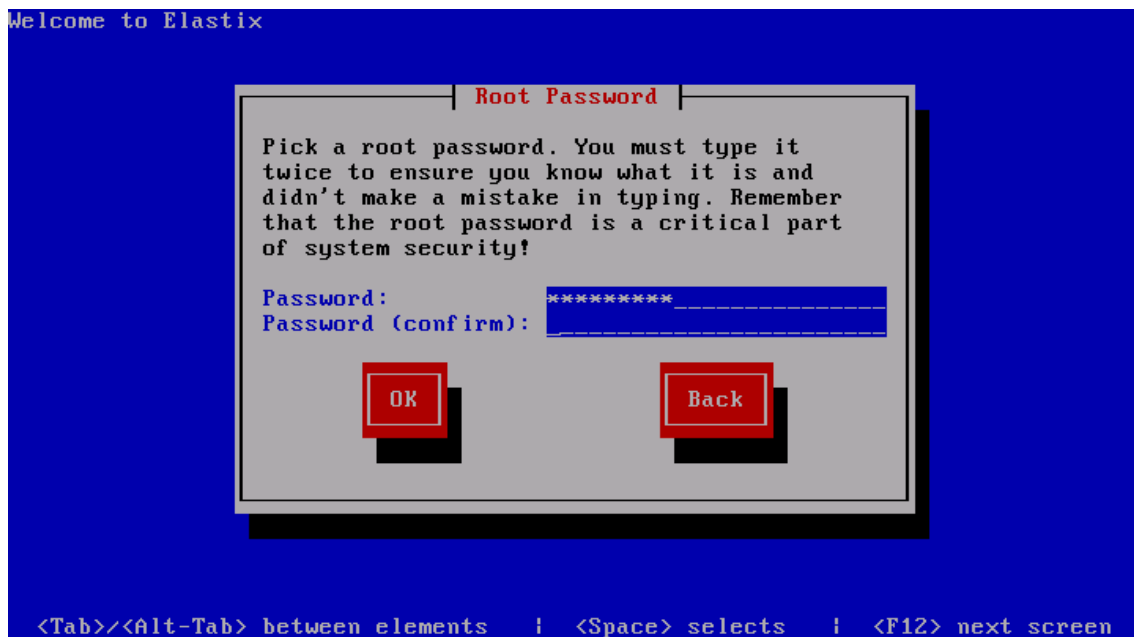
Selección de tipo de teclado

Seleccione la hora zona horaria de su región.



Selección de zona horaria

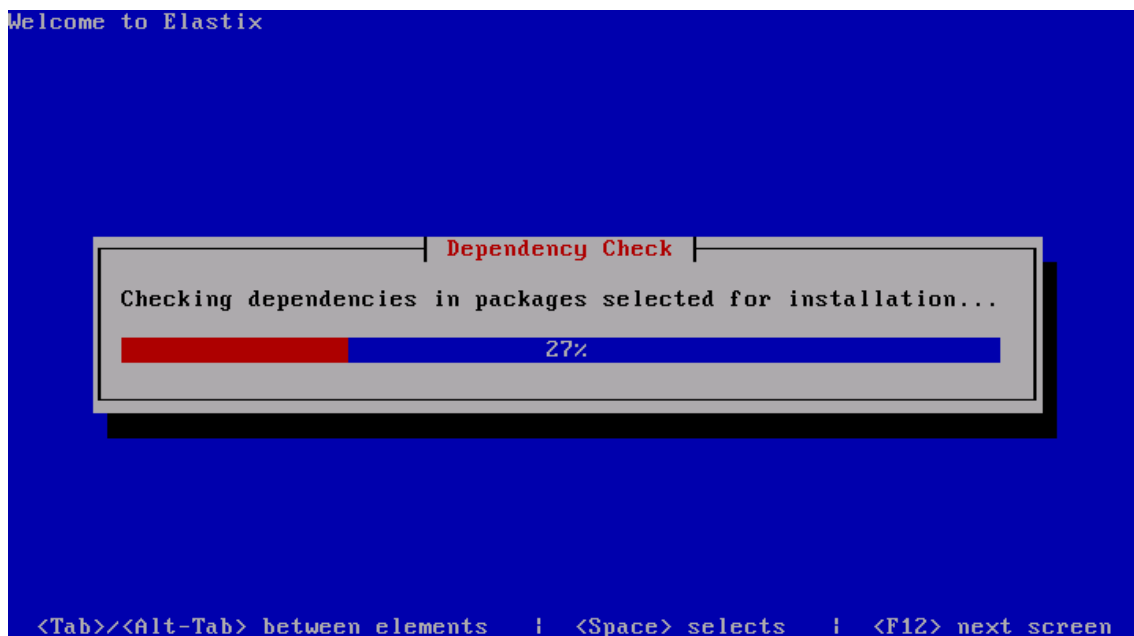
Digite la contraseña que será usada por el administrador de Elastix. Recuerde que esta es una parte crítica para la seguridad del sistema.



Escogiendo la contraseña de root

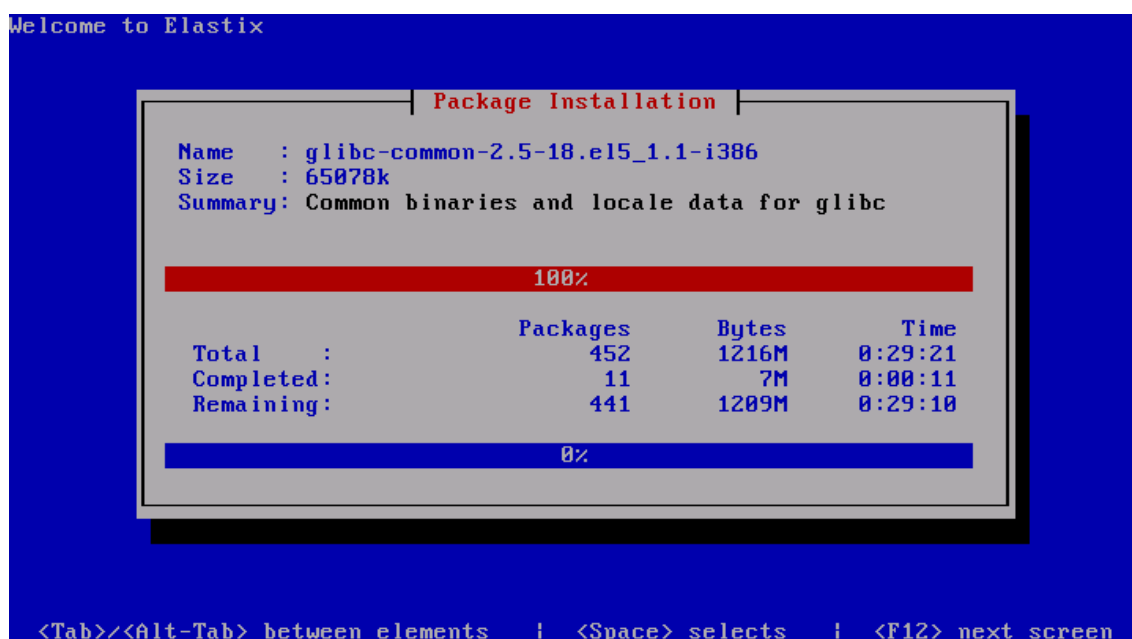
***Nota:** Los procedimientos a continuación los realizará el CD de instalación de manera automática.*

Primero se buscará las dependencias necesarias para la instalación.



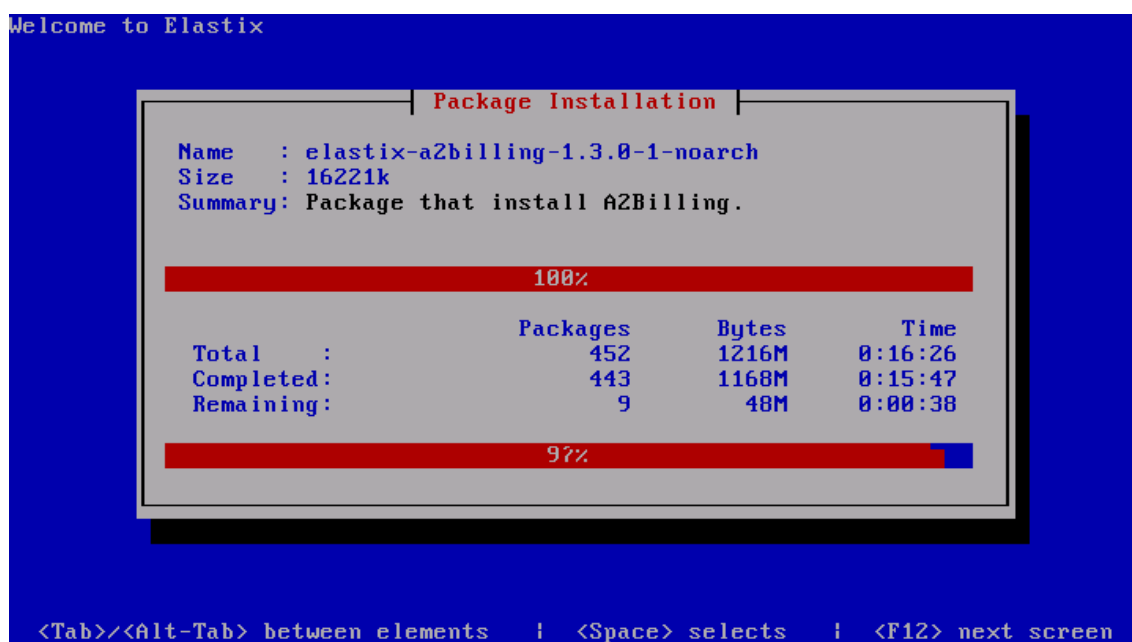
El instalador se encuentra revisando dependencias entre paquetes

Luego se procede con la instalación, inicialmente usted verá algo como lo siguiente.



Inicio del proceso de instalación de paquetes

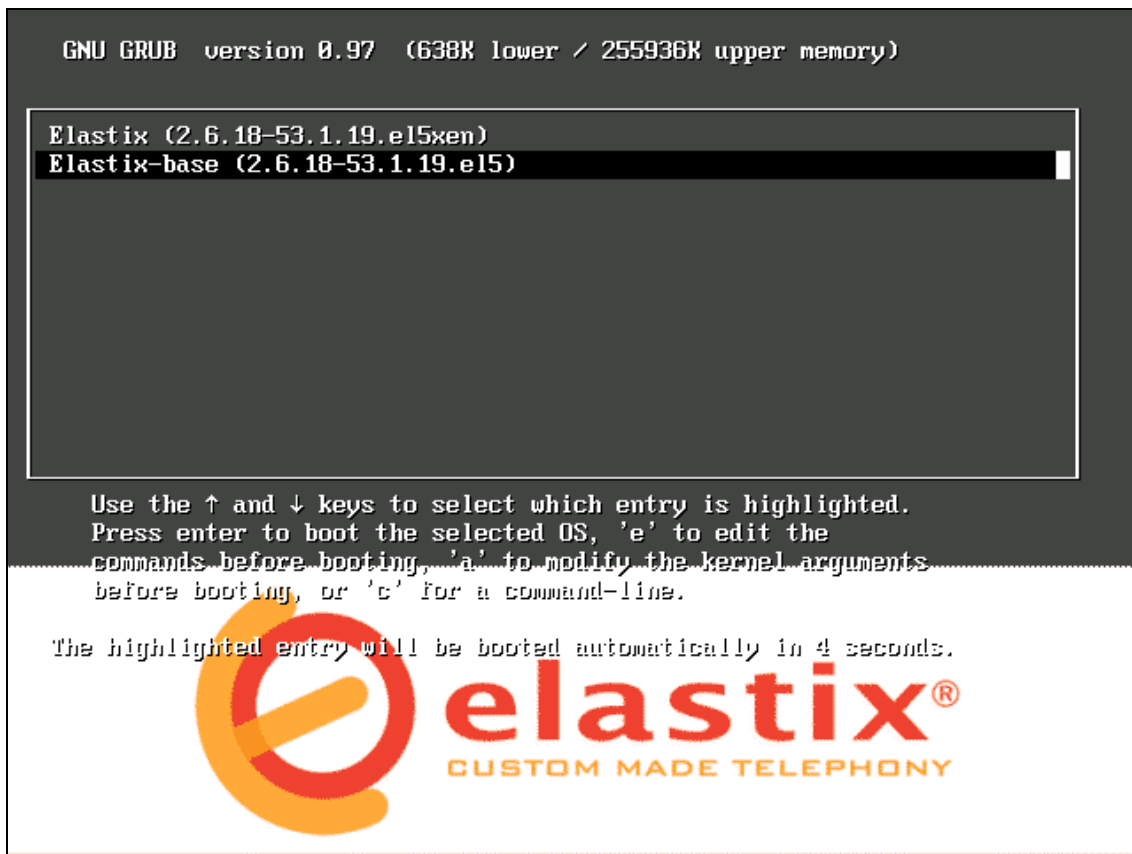
Imagen del proceso de instalación por finalizar.



Fin del proceso de instalación de paquetes

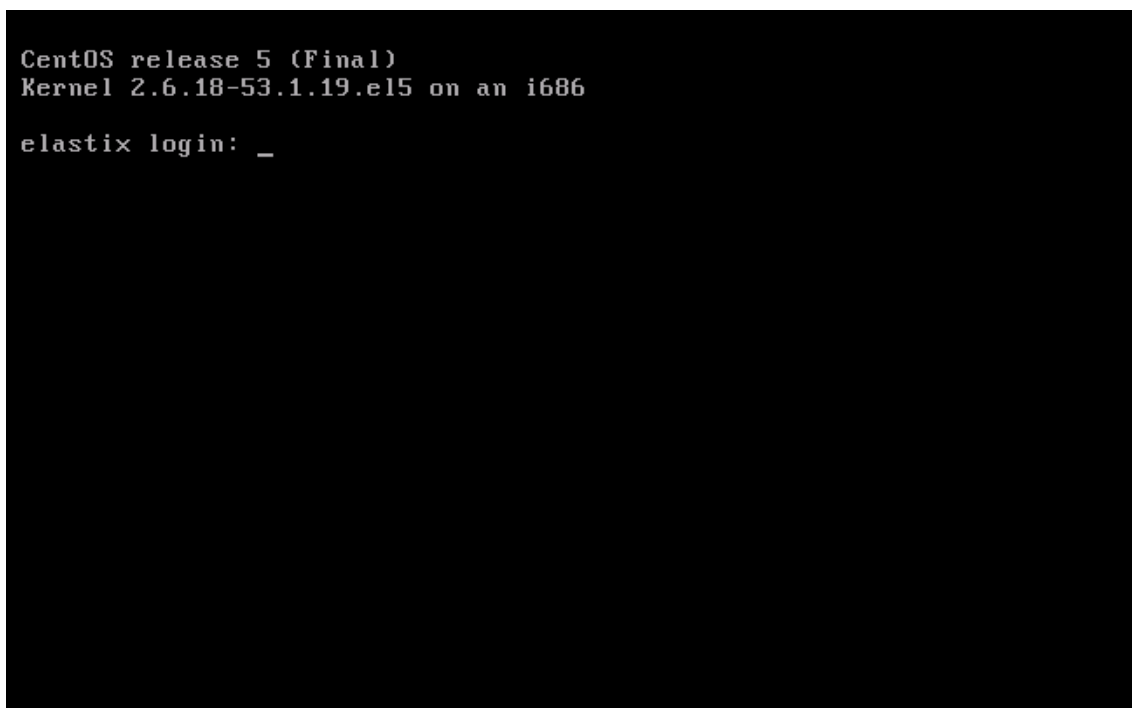
Una vez se realice la instalación completa, se procede a reiniciar el sistema.

Luego de reiniciar el sistema usted podrá escoger entre las opciones de boot la distro de Elastix.



Pantalla de GRUB para seleccionar una imagen de arranque

Ingresa como usuario `root` y la contraseña digitada al momento de la instalación.



Pantalla de autenticación para ingresar a Elastix desde consola

5.2 Instalando Elastix sobre un CentOS o Red Hat preinstalado

A pesar de que el CD de Elastix es la manera recomendada de instalar la distro existen algunos escenarios donde es deseable partir de un sistema operativo pre-instalado y sobre él instalar. Algunas razones para hacer esto podrían ser las siguientes.

- Soporte para 64 bits y obviamente más de 4 GB de RAM
- Para obtener soporte Red Hat

Para cumplir con este cometido se ha ideado un script llamado CentOS2Elastix. Este *script* básicamente convierte un CentOS en un Elastix. CentOS es una popular distribución de Elastix basada en la popular también Red Hat. Por lo dicho, al menos en teoría, este *script* también debería funcionar sobre Red Hat.

Sin embargo, no todo es soplar y hacer botellas. Solo ciertas versiones de CentOS son soportadas. La última versión del *script* CentOS2Elastix funciona con CentOS 5.2 usando kernel 2.6.18-53. Es necesario que se encuentre instalado este kernel exactamente pues algunos *drivers* de Elastix son dependientes del kernel. Por ejemplo, Zaptel.

El script CentOS2Elastix se lo puede descargar de elastix.org. Sin embargo, se trata realmente de una imagen ISO que hay que “montar” antes de poder acceder a su contenido. Por esta razón, luego de descargar la imagen podemos seguir el siguiente procedimiento.

```
# mkdir /testFolder
# mount -o loop ElastixCore-xxxx.iso /testFolder
# cd /testFolder
# ./install
```

En el ejemplo anterior debemos reemplazar las letras xxxx por la versión del *script* que nos hemos descargado.

Nota: Para más información existe un pequeño how-to en el wiki de Elastix.org

5.3 Instalando sobre un *soft* RAID

Una configuración muy útil que Elastix ha heredado de Linux es la posibilidad de configurar arreglos de disco por software (Software RAID), permitiendo agregar un nivel mínimo de redundancia a un bajo costo.

Aunque están soportados 3 tipos de RAID (0, 1 y 5), el que generalmente se utiliza es del RAID 1 o “espejo” que nos permite tener duplicada la información de las particiones. Es importante recalcar que no es estrictamente necesario que todas las particiones sean agregadas al arreglo, sin embargo, es recomendable que si se tienen 2 discos duros, se haga un espejo de todas las particiones.

Un esquema de particionamiento común consiste en tener una partición de 100Mb para “/boot”, una partición de 2Gb para la memoria *swap* y el resto del espacio en una partición para la raíz “/”. Si se sigue este esquema pero con un arreglo de tipo 1 se deberán crear las 3 particiones en ambos discos duros de tal forma que al final se tendrán 6 particiones de tipo RAID.

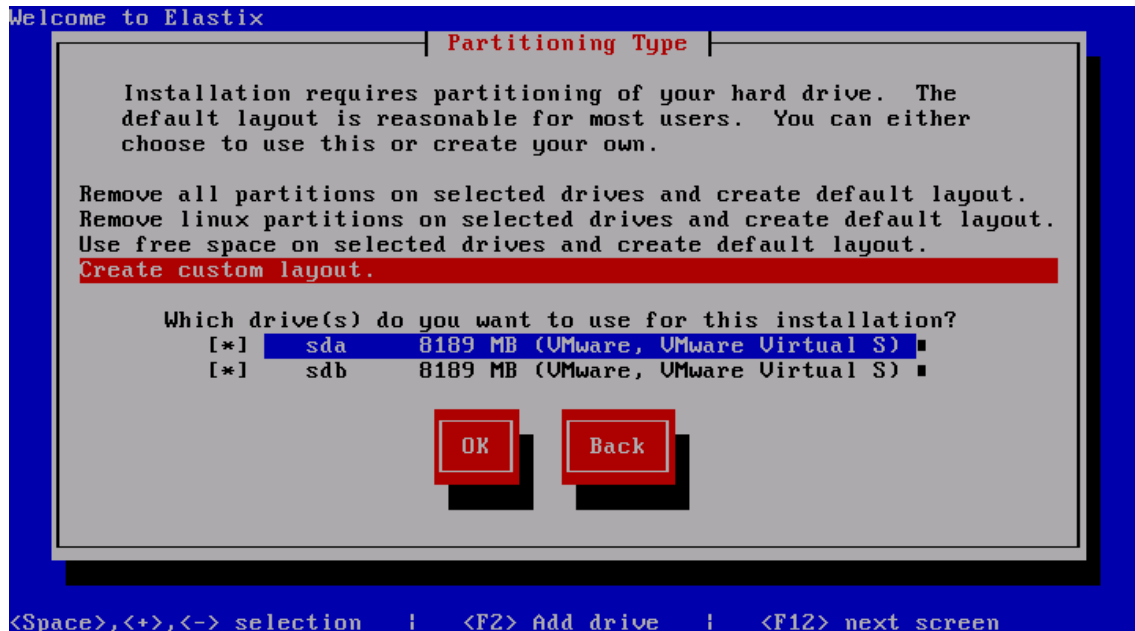
La configuración del Software RAID se la hace durante el proceso de instalación del Elastix, pero utilizando el modo avanzado. Básicamente se presentará un paso adicional correspondiente al particionamiento de los discos duros, éste paso se subdivide a su vez en pasos adicionales que son la creación de las particiones de tipo RAID y luego la creación de los dispositivos RAID que agrupan a las particiones creadas anteriormente.

Para ingresar en modo avanzado es necesario digitar el comando `advanced` en la pantalla inicial.



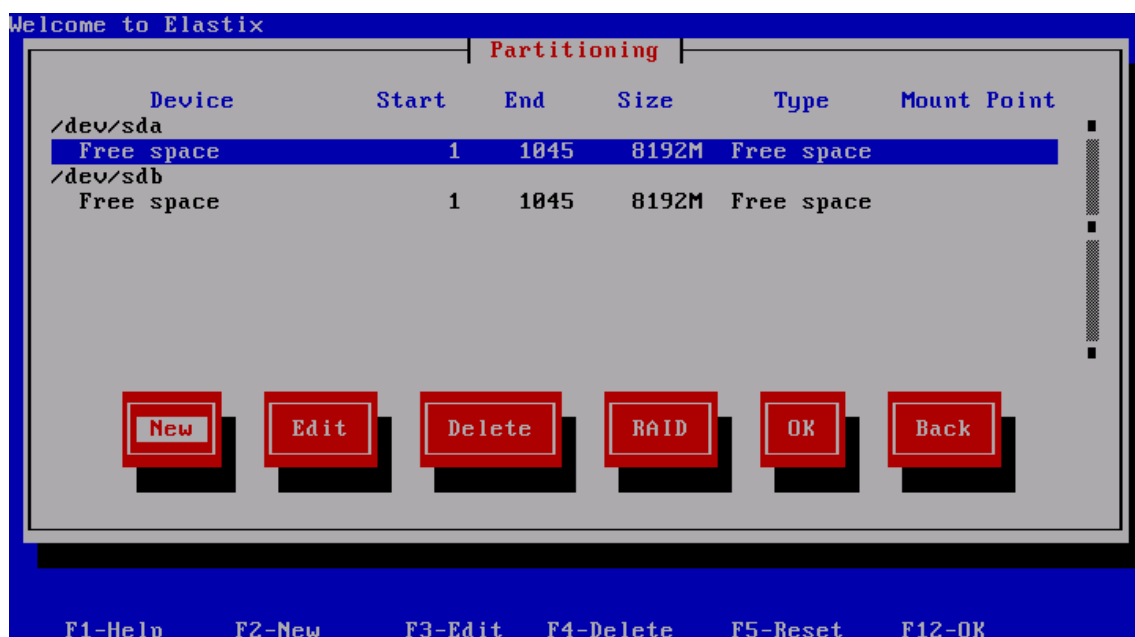
Pantalla de instalación inicial: escogiendo opción “advanced”

Al igual que en el proceso de instalación normal se deberá seleccionar el tipo de teclado de acuerdo al idioma, luego de lo cual aparecerá una pantalla preguntando el tipo de particionamiento que se va a realizar.



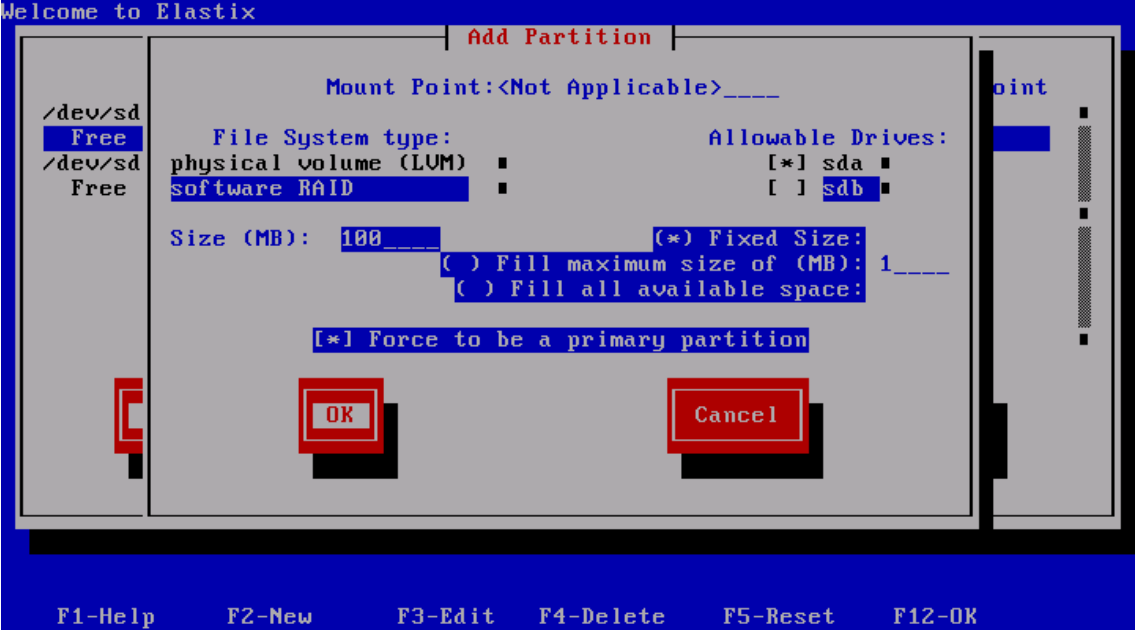
Opciones de tipo de particionamiento

Se debe elegir la opción “*Create custom layout*” (Crear un diseño personalizado) y asegurarse de que estén listados y seleccionados los dos discos duros del servidor. La pantalla de particionamiento mostrará el listado de discos duros con las particiones que se vayan creando.



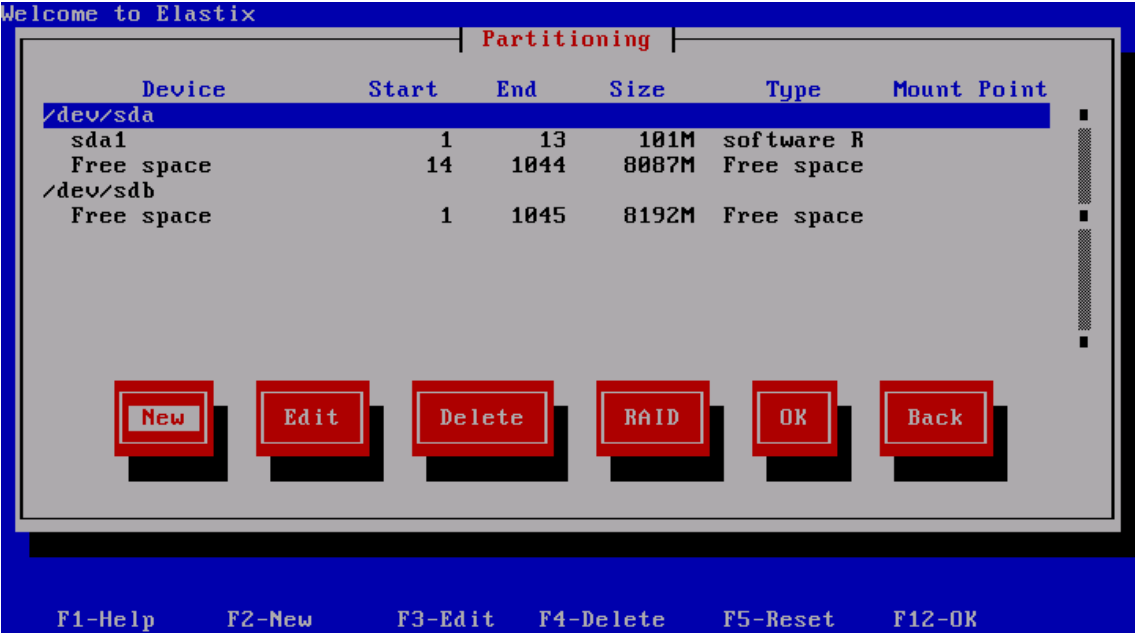
Herramienta de particionamiento mostrando los discos duros aun sin particionar

El primer paso será crear todas las particiones de tipo RAID, para ello se elige la opción “New”.



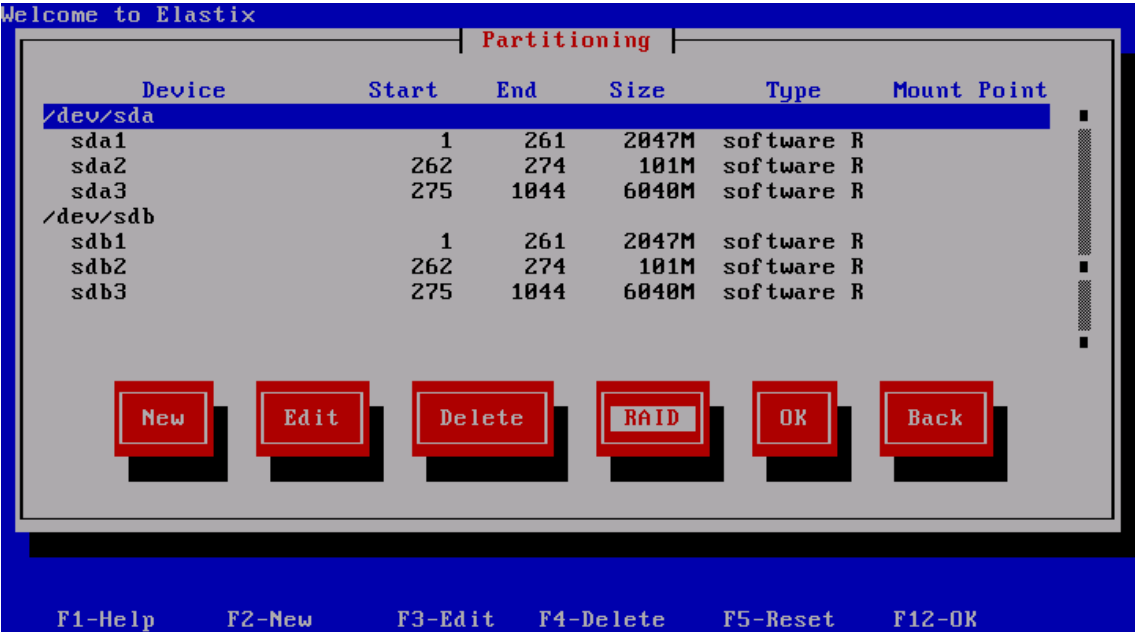
Creando una partición

En el tipo de sistema de archivos se debe elegir “*software RAID*”, adicionalmente se debe especificar el disco duro en el que se creará la partición, el tamaño de la misma y si será una partición primaria. Luego de seleccionar “OK” la nueva partición aparecerá en el listado como se muestra a continuación.



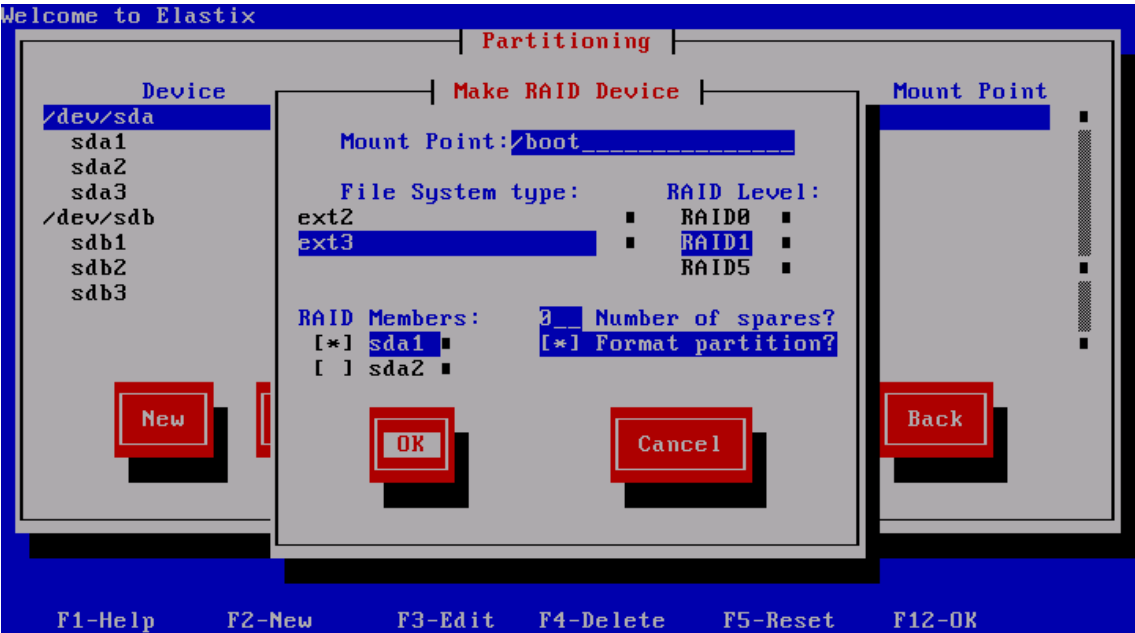
Resumen del particionamiento hasta este punto

Se debe repetir este proceso por cada una de las particiones RAID y finalmente se tendrá un listado como el que se muestra a continuación.



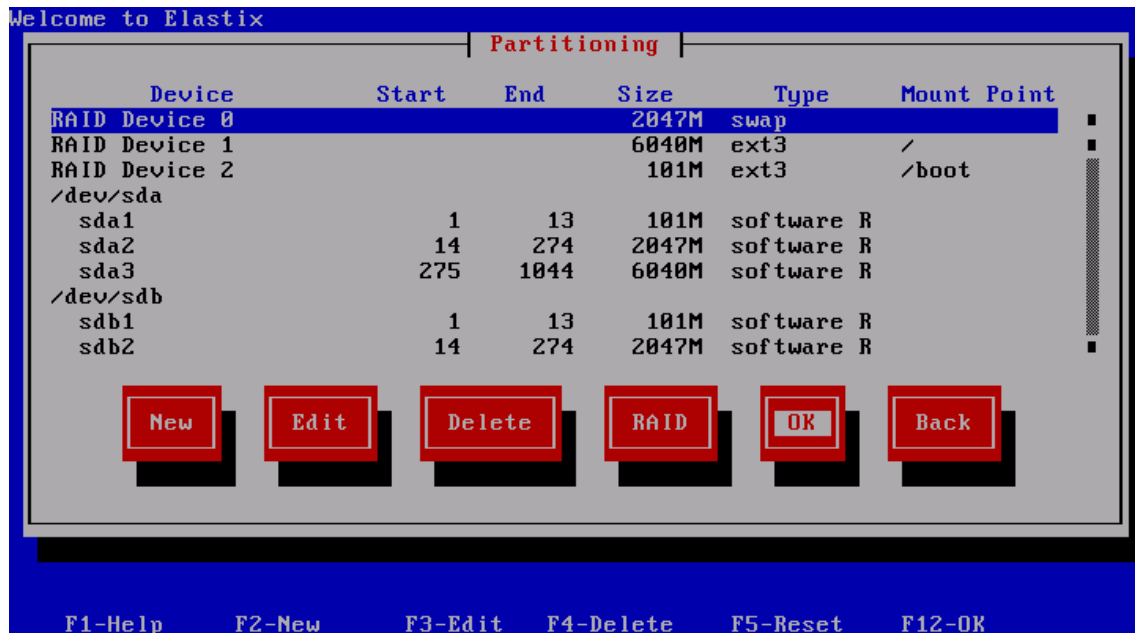
Listado de particiones al final del proceso

En este listado se aprecia que en el particionamiento es exactamente igual entre los dos discos duros. El siguiente paso será la creación de los dispositivos RAID. Para ello se selecciona la opción de “RAID”.



Creación del RAID

En esta pantalla se debe elegir el punto de montaje que tendrá el dispositivo, el tipo de sistema de archivos, el tipo de RAID (1) y las particiones RAID que le pertenecen. Hay que tener mucho cuidado de seleccionar adecuadamente las particiones correspondientes. Este proceso se repetirá por cada uno de los dispositivos RAID de acuerdo al esquema de particionamiento definido. Posteriormente, los dispositivos RAID aparecerán en el listado de las particiones.



Finalmente hemos terminado, el RAID está creado

Finalmente, se elige la opción “OK” y se sigue con el procedimiento de instalación normal.

6

Elastix Overview

Los que dicen "imposible" no deberían interrumpir a los que están intentando

-- Thomas Edison

Elastix es una distribución de "Software Libre" de Servidor de Comunicaciones Unificadas que integra en un solo paquete algunas tecnologías de comunicaciones claves como:

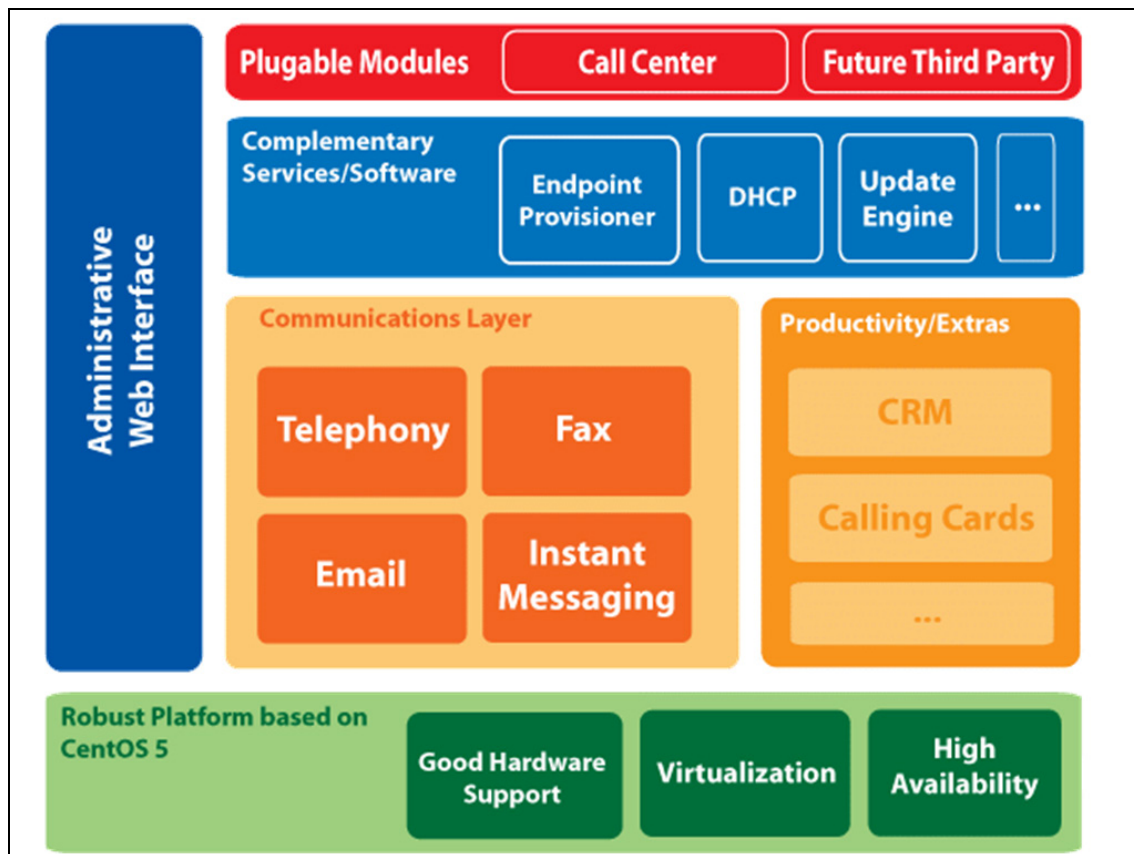
- VoIP PBX
- Fax
- Mensajería Instantánea
- Email
- Colaboración

Al decir distribución nos referimos al concepto de *distro*, es decir un conjunto de paquetes de software que se distribuyen juntos en un mismo medio, en este caso un CD, incluyendo el instalador y sistema operativo. Al final de la instalación tendremos un Servidor de Comunicaciones Unificadas listo para producción.

Elastix implementa gran parte de su funcionalidad sobre 4 programas de software muy importantes como son Asterisk, Hylafax, Openfire y Postfix. Estos brindan las funciones de PBX, Fax, Mensajería Instantánea e Email, respectivamente. La parte de

sistema operativo se basa en CentOS, una popular distribución Linux orientada a servidores.

A continuación una figura un poco más ordenada donde se pueden observar los componentes de Elastix y su relación entre sí.



Esquema general de los componentes de Elastix

6.1 Breve historia del proyecto

Elastix fue creado y actualmente es mantenido por la compañía ecuatoriana PaloSanto Solutions. Elastix fue liberado por primera vez en Marzo de 2006 pero no se trataba de una distro sino más bien de una interfase para mostrar registros de detalles de llamadas para Asterisk, fue recién a finales de Diciembre de 2006 cuando se lo lanzó como una distribución que contenía muchas herramientas interesantes administrables bajo una misma interfase Web que llamó la atención por su usabilidad.

Desde entonces hasta la fecha esta distribución no ha parado de crecer en popularidad y actualmente es una de las preferidas del mercado. En 2007 el proyecto estuvo nominado en 2 categorías para los premios CCA de SourceForge.

6.2 Características de Elastix

Es difícil enlistar todas las características de Elastix en un simple listado, pero las más importantes son según el sitio www.elastix.org y basado en Elastix 1.2:

VoIP PBX

- Grabación de llamadas con interface vía Web
- Voicemails con soporte para notificaciones por email
- IVR configurable y bastante flexible
- Soporte para sintetización de voz
- Herramienta para crear lotes de extensiones lo cual facilita instalaciones nuevas
- Cancelador de eco integrado
- Provisionador de teléfonos vía Web. Esto permite instalar numerosos teléfonos en muy corto tiempo.
- Soporte para Video-fonos
- Interface de detección de hardware de telefonía
- Servidor DHCP para asignación dinámica de IPs a IP-Phones
- Panel de operador. Desde donde el operador puede ver toda la actividad telefónica de manera gráfica y realizar sencillas acciones drag-n-drop como transferencias, parqueos, etc
- Parqueo de llamadas
- Reporte de detalle de llamadas (CDRs) con soporte para búsquedas por fecha, extensión y otros criterios
- Tarifación con reportación de consumo por destino
- Reporte de uso de canales por tecnología (SIP, ZAP, IAX, Local, H323)
- Soporte para colas de llamadas
- Centro de conferencias. Desde donde se puede programar conferencias estáticas o temporales.
- Soporta protocolo SIP, IAX, H323, MGCP, SKINNY entre otros
- Codecs soportados: ADPCM, G.711 (A-Law & μ -Law), G.722, G.723.1 (pass through), G.726, G.729 (si se compra licencia comercial), GSM, iLBC
- Soporte para interfaces análogas FXS/FXO
- Soporte para interfaces digitales E1/T1/J1 a través de protocolos PRI/BRI/R2
- Soporte para interfaces bluetooth para celulares (canal chan_mobile)
- Identificación de llamadas
- Troncalización

- Rutas entrantes y salientes las cuales se pueden configurar por coincidencia de patrones de marcado lo cual da mucha flexibilidad
- Soporte para *follow-me*
- Soporte para grupos de ringado
- Soporte para *paging* e *intercom*. El modelo de teléfono debe soportar también esta característica
- Soporte para condiciones de tiempo. Es decir que la central se comporte de un modo diferente dependiendo del horario
- Soporte para PINes de seguridad
- Soporte DISA
- Soporte *Callback*
- Editor Web de archivos de configuración de Asterisk
- Acceso interactivo desde el Web a la consola de Asterisk

Fax

- Servidor de Fax administrable desde Web
- Visor de Faxes integrado, pudiendo descargarse los faxes desde el Web en formato PDF.
- Aplicación fax-a-email
- Personalización de faxes-a-email
- Control de acceso para clientes de fax
- Puede ser integrado con WinprintHylafax. Esta aplicación permite, desde cualquier aplicación Windows, enviar a imprimir un documento y este realmente se envía por fax.
- Configurador Web de plantillas de e-mails

General

- Ayuda en línea embebida
- Elastix está traducido a 20 idiomas
- Monitor de recursos del sistema
- Configurador de parámetros de red
- Control de apagado/re-encendido de la central vía Web
- Manejo centralizado de usuarios y perfiles gracias al soporte de ACLs
- Administración centralizada de actualizaciones
- Soporte para backup/restore a través del Web
- Soporte para temas o *skins*

- Interface para configurar fecha/hora/uso horario de la central

Email

- Servidor de Email con soporte multi-dominio
- Administrable desde Web
- Interfase de configuración de Relay
- Cliente de Email basado en Web
- Soporte para "cuotas" configurable desde el Web

Colaboración

- Calendario integrado con PBX con soporte para recordatorios de voz
- Libreta telefónica (*Phone Book*) con capacidad *clic-to-call*
- Dos productos de CRM integrados a la interfase como vTigerCRM y SugarCRM

Extras

- Interface de generación de tarjetas de telefonía basada en software A2Billing
- CRM completo basado en el producto vTigerCRM
- También versión open source de SugarCRM

Call Center

- Módulo de call center con marcador predictivo incluido. Más detalle de este módulo más abajo.

Mensajería instantánea

- Servidor de mensajería instantánea basado en OpenFire e integrado a PBX con soporte para protocolo Jabber, lo que permite usar una amplia gama de clientes de IM disponibles
- Se puede iniciar una llamada desde el cliente de mensajería (si se usa el cliente Spark)
- El servidor de mensajería es configurable desde Web
- Soporta grupos de usuarios

- Soporta conexión a otras redes de mensajería como MSN, Yahoo Messenger, GTalk, ICQ, etc. Esto permite estar conectado a varias redes desde un mismo cliente
- Reporte de sesiones de usuarios
- Soporte para *plugins*
- Soporta LDAP
- Soporta conexiones server-to-server para compartir usuarios

6.3 Licenciamiento

Elastix es software libre distribuido bajo licencia GPL versión 2. Es decir que puede ser usado, copiado, estudiado, modificado y redistribuido libremente según los lineamientos de esta licencia.

Para acceder al texto completo de la licencia el lector puede seguir el siguiente vínculo.

<http://www.gnu.org/licenses/old-licenses/gpl-2.0.txt>

6.4 Sobrevuelo por la interfase de administración Web de Elastix

La interfase Web de Elastix es una aplicación completa de administración del servidor de comunicaciones unificadas escrita en su mayoría en lenguaje PHP.

En este capítulo haremos un sobrevuelo de la funcionalidad mas importante que podemos administrar desde esta interfase y a continuación explicaremos como se encuentran organizados los menús y módulos con los que cuenta Elastix. Cada menú aparecerá en negritas encabezando una tabla con los módulos que dicho menú contiene.

System

| Nombre módulo | Descripción |
|---------------|---|
| Dashboard | <i>Dashboard</i> es una especie es una especie de escritorio donde el usuario puede ver un resumen de actividad en Elastix como sus últimos emails, sus <i>voicemails</i> , sus últimos faxes, si tiene algo agendado en el calendario, etc |

| | |
|------------------------|---|
| System Info | Muestra información de sistema como uso de memoria, CPU y disco duro |
| Network | Menú de configuración de parámetros de red |
| Network Parameters | Aquí se pueden configurar parámetros de red como dirección IP y máscara de red, <i>gateway</i> , nombre de <i>host</i> , servidores DNS, entre otros. |
| DHCP Server | Permite configurar el servidor DHCP embebido que viene con Elastix |
| User Management | Menú de administración de usuarios de Elastix |
| Groups | Permite configurar grupos de usuarios |
| Users | Permite administrar usuarios y asignarlos a grupos. También permite asociar cuentas de <i>email</i> y extensiones telefónicas a usuarios |
| Group Permission | Aquí se configuran los permisos de acceso a los diferentes módulos para un grupo determinado |
| Load Module | Permite cargar un módulo de Elastix |
| Shutdown | Sirve para apagar el servidor |
| Hardware Detection | Módulo de detección de hardware telefónico |
| Updates | Menú de actualizaciones |
| Packages | Listado de paquetes con la opción de instalar o actualizar |
| Repositories | Se pueden configurar los repositorios en base a los cuales se realizan las actualizaciones |
| Backup/Restore | Módulo para respaldar el servidor Elastix y también para subir respaldos y restituir información |
| Preferences | Menú para configurar preferencias varias |
| Language | Cambia el idioma de toda la interfase Web de Elastix |
| Date/Time | Cambia la fecha, hora y zona horaria del servidor |
| Themes | Permite cambiar los temas (<i>skins</i>) para darle a la interfase de Elastix un diferente <i>look</i> |

PBX

| Nombre módulo | Descripción |
|------------------------|---|
| PBX Configuration | Aquí se encuentra embebido freePBX. Desde aquí se hacen la mayoría de configuraciones a nivel de central telefónica |
| Flash Operator Panel | Panel de operador basado en flash, herramienta muy útil para el recepcionista |
| Voicemails | Listado de <i>voicemails</i> . Se debe haber asociado previamente al usuario con una extensión telefónica para poder ver el listado |
| Monitoring | Listado de grabaciones telefónicas. Al igual que con el módulo anterior el usuario debe estar asociado con una extensión |
| Echo Cancellor | Actividad del cancelador de eco |
| Endpoint Configuration | Herramienta muy útil para provisionar lotes grandes de teléfonos en corto tiempo |
| Conference | Módulo para agendar conferencias temporales |
| Extensions Batch | Módulo para crear grandes lotes de extensiones |
| Tools | Menú con herramientas varias |
| Asterisk CLI | Permite ejecutar comandos del CLI desde el Web |
| File Editor | Permite editar archivos de texto plano desde el Web |

Fax

| Nombre módulo | Descripción |
|------------------|--|
| Virtual Fax List | Listado de extensiones de fax virtuales. Es decir que recibirán faxes en formato PDF en un buzón de <i>email</i> |
| New Virtual Fax | Este módulo permite crear extensiones de fax nuevas |
| Fax Master | Permite configurar una dirección <i>email</i> que recibirá notificaciones del funcionamiento del fax |
| Fax Clients | Configuración de permisos de acceso para |

| | |
|----------------|---|
| | aplicaciones clientes de fax |
| Fax Visor | Visor de faxes que permite visualizar faxes en formato PDF |
| Template Email | Herramienta de configuración de plantilla de <i>email</i> que se enviará cada vez que arribe un fax |

Email

| Nombre módulo | Descripción |
|---------------|---|
| Domains | Creación de dominios de <i>email</i> . Elastix soporta multidominios. |
| Accounts | Creación de cuentas de email y asignación de cuotas de espacio en disco duro |
| Relay | Configuración de <i>relay</i> para permitir a otras redes utilizar a Elastix para enviar su email |
| Webmail | Interfase de Webmail basada en software Roundcube |

IM

| Nombre módulo | Descripción |
|---------------|--|
| OpenFire | Interfase embebida para administrar el servidor Openfire |

Reports

| Nombre módulo | Descripción |
|----------------|---|
| CDR Report | Reporte de CDRs con opciones de filtrado por campos y por fechas |
| Channels Usage | Reporte de uso de canales. Se pueden ver gráficos por diferentes tipos de tecnología como SIP e IAX |
| Billing | Menú de tarificación |
| Rates | Aquí se pueden establecer las tarifas dependiendo de la troncal y el prefijo telefónico |
| Billing Report | Reporte de tarificación con filtrado por fechas y campos. Básicamente se calcula |

| | |
|--------------------------|--|
| | y muestra el costo de cada llamada |
| Destination Distribution | Gráfico de pastel de la distribución por destinos. Hay 3 criterios: por costo, por número de llamadas y por tiempo de duración de las llamadas |
| Billing Setup | Configuración de las troncales habilitadas para la tarificación. También se establece aquí la tarifa por omisión |
| Asterisk Logs | Interfase para ver el <i>log</i> de Asterisk con filtrado por fechas y cadenas de texto |

Extras

| Nombre módulo | Descripción |
|-------------------|--|
| vTigerCRM | Software de poderoso CRM embebido |
| Calling Cards | Interfase basada en software A2Billing para administrar tarjetas de llamadas |
| Downloads | Menú de descargas |
| Softphones | Listado de aplicaciones de <i>softphones</i> recomendadas |
| Fax Utilities | Listado de aplicaciones de fax recomendadas |
| Instant Messaging | Listado de clientes de IM recomendados |
| SugarCRM | Software CRM en su versión de código abierto |

Agenda

| Nombre módulo | Descripción |
|---------------|---|
| Calendar | Módulo de calendario para agendar eventos que inclusive pueden generar llamadas telefónicas automáticas |
| Address Book | Libreta de direcciones |
| Recordings | Interfase para grabar mensajes que se pueden asociar con el módulo Calendar y que se reproducen cuando se genera una llamada automática |

6.5 Elastix.org: Sitio oficial del proyecto

El principal recurso al que puede acceder el usuario en busca de información acerca de Elastix es el sitio Web oficial del proyecto ubicado en <http://www.elastix.org>

Aquí el usuario puede encontrar algunas herramientas útiles como:

- Foros
- Wiki
- Chat en vivo
- Suscripción a listas de correo
- Links a descargas
- entre otra información

Foros

Una de las herramientas más importantes en el sitio Web de Elastix son los foros. Los foros son muy activos y para participar en ellos basta registrarse gratuitamente como usuario. Sin duda esta es una buena manera de buscar solución a problemas comunes y los desarrolladores de Elastix suelen visitar esta herramienta a la caza de posibles bugs. Sin embargo, el foro es un lugar público así que antes de participar hay que tener en cuenta algunas reglas de etiqueta de foros básicas.

- Antes de preguntar en el foro busque si su problema ha sido resuelto antes. La mayoría de problemas comunes ya han sido reportados antes. Si vuelve a preguntar es posible que no le contesten adecuadamente, recuerde que es un foro público.
- Los foros se encuentran organizados de cierta manera. Se debe tratar en todo momento de añadir un nuevo comentario a la categoría correcta, caso contrario se puede “ensuciar” el foro y también el nuevo comentario podría ser ignorado.
- Use un lenguaje adecuado para no herir la susceptibilidad de nadie. La amabilidad es un buen ingrediente si desea que otras personas colaboren con su problema.
- El foro no es un espacio publicitario.
- El foro es un espacio moderado. No incluya lenguaje subido de tono, amenazas, comentarios mal intencionados o que puedan herir la susceptibilidad de alguien. Su mensaje puede ser eliminado permanentemente.

- No adjunte imágenes o archivos pesados. Si necesita subir un archivo mayor a 2 MB trate de subirlo a otro servidor y simplemente copie el link. Nos gustaría aceptar todos los archivos pero debemos cuidar los recursos de los servidores para poder servir a la mayor cantidad de usuarios posible.
- Use un título de mensaje relevante y si en algún momento la discusión cambia de tópico abra otro mensaje para tratar el nuevo tema. Esto ayudará a tener el foro más legible.

7

Elastix: Configuración Básica

Educación es elevar al hombre al nivel de su tiempo

-- José Martí

7.1 Ingreso al Sistema

El usuario y contraseña por defecto para ingreso al sistema son los siguientes:

Usuario: admin

Contraseña: palosanto

7.2 Configuración de red

Lo primero que debemos configurar luego de ingresar al equipo son los parámetros de red:

La opción “Red” del Menú “Sistema” del Elastix nos permite visualizar y configurar los parámetros de red del servidor.

Network Parameters

Edit Network Parameters
* Required field

Host (Ex. host.example.com): elastix.example.com

Primary DNS: * dns.example.com

Default Gateway: * 192.168.8.1

Secondary DNS:

Ethernet Interfaces List

Start Previous (1 - 1 of 1) Next End

| Device | Type | IP | Mask | MAC Address | HW Info | Status |
|----------------------------|--------|--------------|---------------|-------------------|---|-----------|
| Ethernet 0 | STATIC | 192.168.8.86 | 255.255.255.0 | 00:13:8F:CC:62:99 | RealTek RTL8139 at 0xf883cc00, 00:13:8f:cc:62:99, IRQ 201 | Connected |

Start Previous (1 - 1 of 1) Next End

Configuración de parámetros de red

- **Host:** Nombre del Servidor, por ejemplo: pbx.example.com
- **Puerta de Enlace:** Dirección IP de la Puerta de Enlace (*Gateway*)
- **DNS Primario:** Dirección IP del Servidor de Resolución de Nombres (DNS) Primario
- **DNS Secundario:** Dirección IP del Servidor de Resolución de Nombres (DNS) Secundario o Alternativo.

7.3 Creando una extensión

Esta sección está dirigida a los teléfonos, *softphones*, sistemas paginadores, o cualquier cosa que pueda ser considerada como una “extensión”.

Definir y corregir extensiones es probablemente la tarea más común realizada por un administrador de PBX, y como tal, se encontrará muy al corriente de esta página. Hay actualmente cuatro tipos de dispositivos o tecnologías soportadas: SIP, IAX2, ZAP y “Custom”.

Para crear una “Nueva extensión” ingrese al Menú “PBX”. Por defecto se accede a la sección “Configuración PBX”, en esta sección escogemos del panel izquierdo la opción “Extensiones”. Ahora podremos crear una nueva extensión.

Primero escoja el dispositivo de entre las opciones disponibles:

Add an Extension

Please select your Device below then click Submit

Device

Device

Generic SIP Device

Generic SIP Device

Generic IAX2 Device

Generic ZAP Device

Other (Custom) Device

Submit

Creando nueva extensión

- **Generic SIP Device:** El SIP es el protocolo estándar para los teléfonos VoIP y ATA. La mayoría de teléfonos IP soportan SIP.
- **Generic IAX2 Device:** IAX es el “protocolo Inter Asterisk Exchange”, un nuevo protocolo apoyado solamente por algunos dispositivos (Por ejemplo, los teléfonos basados en PA1688, y el IAXy ATA).
- **Generic ZAP Device:** ZAP es un dispositivo de hardware conectado al servidor Elastix. Por lo general tarjetería PCI controlada con los *drivers* del proyecto Zaptel (de allí el nombre de ZAP).
- **Other (Custom) Device:** Custom nos permite escribir directamente una entrada en los archivos de configuración y por ende esta entrada debe estar en formato de extensión entendible por Asterisk. Puede también ser utilizado para “mapear” una extensión a un número “externo”. Por ejemplo, para enrutar la extensión 211 a 1-800-555-1212, se puede crear una extensión “Custom” 211 y en la caja de texto del “dial” se puede ingresar: Local/18005551212@outbound-allroutes.

Una vez haya escogido el dispositivo correcto, de clic en Ingresar.

Luego de escoger el tipo de dispositivo nos aparecerá un formulario que varía un poco dependiendo de lo que hayamos escogido previamente. Nosotros supondremos que el usuario ha escogido SIP pues es lo más común.

Add SIP Extension

Add Extension

| | | | |
|----------------------------------|--|---------------------------------------|---|
| User Extension | <input type="text"/> | Privacy | |
| Display Name | <input type="text"/> | Privacy Manager | No <input type="button" value="v"/> |
| CID Num Alias | <input type="text"/> | Dictation Services | |
| SIP Alias | <input type="text"/> | Dictation Service | Disabled <input type="button" value="v"/> |
| Extension Options | | Dictation Format | Ogg Vorbis <input type="button" value="v"/> |
| Direct DID | <input type="text"/> | Email Address | <input type="text"/> |
| DID Alert Info | <input type="text"/> | Recording Options | |
| Music on Hold | acc_1 <input type="button" value="v"/> | Record Incoming | On Demand <input type="button" value="v"/> |
| Outbound CID | <input type="text"/> | Record Outgoing | On Demand <input type="button" value="v"/> |
| Ring Time | Default <input type="button" value="v"/> | Voicemail & Directory | |
| Call Waiting | Disable <input type="button" value="v"/> | Status | Disabled <input type="button" value="v"/> |
| Emergency CID | <input type="text"/> | Voicemail Password | <input type="text"/> |
| Device Options | | Email Address | <input type="text"/> |
| This device uses sip technology. | | Pager Email Address | <input type="text"/> |
| secret | <input type="text"/> | Email Attachment | <input type="radio"/> yes <input checked="" type="radio"/> no |
| dtmfmode | rfc2833 | Play CID | <input type="radio"/> yes <input checked="" type="radio"/> no |
| Fax Handling | | Play Envelope | <input type="radio"/> yes <input checked="" type="radio"/> no |
| Fax Extension | FreePBX default <input type="button" value="v"/> | Delete Vmail | <input type="radio"/> yes <input checked="" type="radio"/> no |
| Fax Email | <input type="text"/> | VM Options | <input type="text"/> |
| Fax Detection Type | None <input type="button" value="v"/> | VM Context | default |
| Pause after answer | 0 | VmX Locator™ | Disabled <input type="button" value="v"/> |
| | | <input type="button" value="Submit"/> | |

Formulario de creación de nueva extensión SIP

Como podemos observar se pueden configurar aquí muchas cosas interesantes pero no todos los datos son necesarios para conseguir una extensión funcional así que explicaremos aquí solo los más importantes.

- **Extensión del Usuario:** Debe ser único. Éste es el número que se puede marcar de cualquier otra extensión, o directamente del recepcionista Digital si está permitido. Puede ser cualquier longitud, pero convencionalmente se utiliza una extensión de tres o cuatro cifras.
- **Display Name:** Es el nombre del Caller ID, para llamadas de este usuario serán fijadas con su nombre. Sólo debe ingresar el nombre no la extensión.

- **Secret:** Esta es la contraseña usada por el dispositivo de la telefonía para autenticar al servidor de Asterisk. Esto es configurado generalmente por el administrador antes de dar el teléfono al usuario, y generalmente no se requiere que lo conozca el usuario. Si el usuario está utilizando un *softphone*, entonces necesitarán saber esta contraseña para configurar su software.

Configuración de teléfono *softphone*

Nota: Aquí explicaremos algo muy breve. Para más detalles visitar el capítulo “Configuración de teléfono IP”.

Al configurar un teléfono *softphone* lo que lograremos es tener una PC conectada que cumpla con las mismas funciones de un teléfono convencional, para esto es necesario instalar un software que haga las veces de teléfono. Además se necesita disponer de audífonos y micrófono. Existen varias alternativas para *softphones*, entre ellos podemos citar los siguientes:

- **Zoiper:** Este software nos permite trabajar con extensiones de tipo SIP e IAX, además es multiplataforma, podemos descargarlo de la siguiente dirección: <http://www.asteriskguru.com>
- **XtenLite:** Este software trabaja con extensiones SIP únicamente, también es multiplataforma, lo podemos descargar de la siguiente dirección: <http://www.xten.com/index.php?menu=download>

7.4 Configuraciones generales

En esta sección se configuran parámetros generales del sistema Elastix en lo que tiene que ver con telefonía. Explicaremos qué significan los parámetros más relevantes.

Asterisk Dial command options:

Asterisk Outbound Dial command options:

Voicemail

Number of seconds to ring phones before sending callers to voicemail:

Extension prefix for dialing direct to voicemail:

Direct Dial to Voicemail message type:

Use gain when recording the voicemail message (optional): ☐

☐ Do Not Play please leave message after tone to caller

Voicemail VmX Locator

Default Context & Pri: context pri

Timeout/#-press default: context exten pri

Loop Exceed default: context exten pri

Timeout VM Msg:

Max Loop VM Msg:

Direct VM Option:

Msg Timeout: seconds

Msg Play: times

Error Re-tries: times

Company Directory

Find users in the Company Directory by:

☐ Play extension number to caller before transferring call

Operator Extension:

Fax Machine

Extension of fax machine for receiving faxes:

Email address to have faxes emailed to:

Email address that faxes appear to come from:

International Settings

Country Indications:

24-hour format:

Security Settings

Allow Anonymous Inbound SIP Calls?

Online Updates

Formulario de modificación de configuraciones generales

Opciones del comando dial: Explicaremos esto más al detalle en breve.

Company Directory: El directorio de una empresa se puede escuchar marcando *411 ó # cuando se escucha un IVR.

Play extension number: En el mensaje que toca: “Por favor manténgase en la línea mientras transfiero su llamada a la extensión XXX”. Es decir, menciona la extensión a donde se va a transferir la llamada por lo tanto el llamante ya sabrá el número de extensión para llamadas futuras.

Security Settings: Permite a anónimas llamadas SIP ser conectadas. El valor por omisión es “no”.

Online updates: Permite revisar por actualizaciones disponibles. El valor por omisión es “no”.

Opciones del comando Dial (*Dial command options*)

La mayoría de opciones son configuradas en “tr”, lo cual significa que la persona que recibe la llamada puede transferirla usando el “*feature*” configurado para esta funcionalidad. La “r” significa que enviará un “*ring*” al llamante y no pasa ningún audio hasta la llamada sea contestada.

Aquí se describen otras opciones:

- **A(X):** Toca un anuncio a la persona llamada, usando x como archivo.
- **D(X):** Envía un específica señal DTMF
- **h:** Permite a la persona llamada colgar enviando el dígito “*”.
- **H:** Permite al llamante colgar enviando el dígito “*”.
- **r:** Envía un ringeado a la parte llamante.
- **T/t:** Permite a la parte (llamante/llamada) transferir llamadas.
- **W/w:** Permitir a la parte (llamante/llamada) habilitar la grabación de llamadas.

7.5 Configuración de colas

Las colas consisten en grupos de usuarios para los cuales se definen ciertas políticas (o estrategias) principalmente en lo que tiene que ver con llamadas entrantes. Las colas pueden ser llamadas como una extensión más. Los usuarios de las colas pueden ser estáticos, llamados “miembros” (es decir que tienen un número de extensión fijo) o

usuarios dinámicos, llamados agentes (es decir que pueden ingresar a la cola desde cualquier extensión).

Una política o estrategia define la manera como se distribuyen las llamadas entrantes a la cola entre los miembros o agentes. Las políticas de cola que soporta Asterisk al momento son las siguientes:

- **ringall:** ring all available channels until one answers (default)
- **roundrobin:** take turns ringing each available interface (deprecated in 1.4, use rrmemory)
- **leastrecent:** ring interface which was least recently called by this queue
- **fewestcalls:** ring the one with fewest completed calls from this queue
- **random:** ring random interface
- **rrmemory:** round robin with memory, remember where we left off last ring pass

Para configurar una cola en Elastix tenemos que ir al Módulo PBX y escoger en el menú llamado “Queues”.

Para añadir una nueva cola deberemos llenar el siguiente formulario.

Add Queue

Add Queue

| | |
|----------------------|-------------------------------|
| Queue Number: | <input type="text"/> |
| Queue Name: | <input type="text"/> |
| Queue Password: | <input type="text"/> |
| CID Name Prefix: | <input type="text"/> |
| Alert Info: | <input type="text"/> |
| Static Agents: | <div></div> |
| Extension Quick Pick | <div>(pick extension) ▼</div> |

Queue Options

| | |
|------------------------|--------------------------|
| Agent Announcement: | <div>None ▼</div> |
| Music on Hold Class: | <div>inherit ▼</div> |
| Ringin Instead of MoH: | <input type="checkbox"/> |
| Max Wait Time: | <div>Unlimited ▼</div> |
| Max Callers: | <div>0 ▼</div> |
| Join Empty: | <div>Yes ▼</div> |
| Leave When Empty: | <div>No ▼</div> |
| Ring Strategy: | <div>ringall ▼</div> |
| Agent Timeout: | <div>15 seconds ▼</div> |
| Retry: | <div>5 seconds ▼</div> |
| Wrap-Up-Time: | <div>0 seconds ▼</div> |
| Call Recording: | <div>No ▼</div> |
| Event When Called: | <div>No ▼</div> |
| Member Status: | <div>No ▼</div> |
| Skip Busy Agents: | <div>No ▼</div> |

Caller Position Announcements

| | |
|---------------------|------------------------|
| Frequency: | <div>0 seconds ▼</div> |
| Announce Position: | <div>No ▼</div> |
| Announce Hold Time: | <div>No ▼</div> |

Formulario para añadir nueva cola

Queue Number:

Este es el número que puede ser marcado desde cualquier extensión para ser puesto en la cola. Este también es el mismo número que se usa cuando se selecciona un destino. En conclusión este número identifica a la cola y debe ser único.

Queue Name:

Un nombre corto para la cola. Este es usado únicamente para propósitos de identificación.

Queue password:

A una cola se le puede poner una clave. Cuando un agente intenta conectarse, deberían ser preguntados por la clave de la cola.

CID name prefix:

Como un agente puede logonearse en varias colas al mismo tiempo, es útil colocarle un prefijo al Caller ID para que cuando el agente vea la llamada entrante en el teléfono sepa a que cola pertenece la llamada.

Static Agents:

Estos son dispositivos que siempre se loguean en la cola. Esto es útil si se tienen siempre los mismo agentes en una cola. Aquí se deben listar los Agentes de la siguiente forma:

A5013, 0

Donde la letra A indica que se trata de un agente y no de una extensión, seguido del número de agente y el valor 0

Agent Announcement:

Este es un anuncio que es tocado al agente antes de que le conecten una llamada. Esto es útil cuando los agentes no tienen Caller ID en su teléfono y les avisa de una llamada entrante.

Music On Hold:

Este es la música que es tocada al llamante mientras está en la cola por un agente que lo atienda.

Max wait time:

Es el número máximo de segundos que un llamador puede esperar en la cola antes de ser sacado de la misma. La acción a tomar después de este tiempo es configurado más abajo.

Max callers:

Es el máximo número de personas permitidas a esperar en la cola. Si este número es alcanzado, las personas adicionales son enviadas al destino configurado en Destino de Fallas.

Join Empty:

Si se desea que los llamantes ingresen a una cola vacía (sin agentes), entonces hay que setear este parámetro a “yes”. Esto no es recomendado.

Leave when empty:

Se setea a “yes” si se quiere sacar a los llamantes que ingresen a una cola en donde no hay agentes. Recomendado poner no.

Ring strategy:

Hay 6 tipos de estrategia de ringeado:

ringall: ringean todos los agentes válidos hasta que alguno responda.

Roundrobin: ringea en orden desde el primer agente válido

leastrecent: ringea al agente cual fue el menos reciente llamado en la cola

fewestcalls: ringea al agente con menos llamadas completadas en la cola.

random: ringea a un agente en forma aleatoria.

Rmemory: round robin con memoria, recuerda donde se conectó la última llamada.

Agent timeout:

El número de segundo que se espera para intentar nuevamente ringear a un agente.

Wrap-up-time:

Después de una llamada exitosa, indica el número de segundos antes de declarar al agente libre y enviarle una nueva llamada, El valor por defecto es 0.

Call Recording:

Opción de grabar las llamadas entrantes a la cola.

Fail Over Destination:

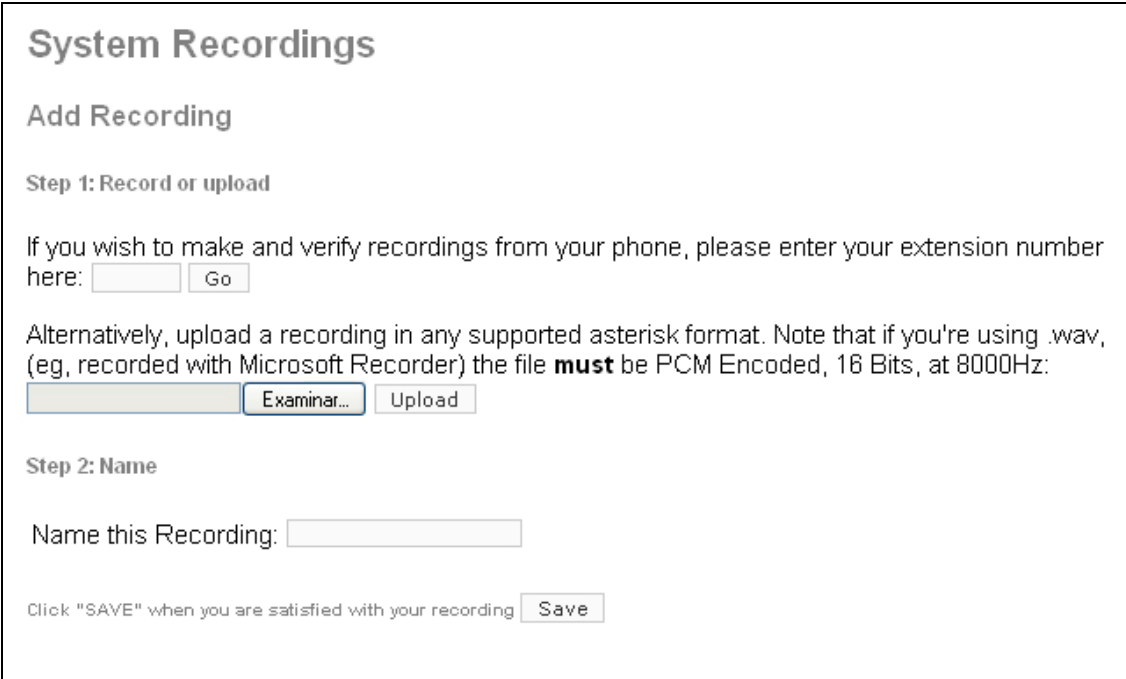
Este el destino si una llamada de la cola no pudo ser atendida.

7.6 IVR y sistema de grabación

Grabación de mensaje de bienvenida

Antes de Crear un IVR es necesario tener un mensaje de bienvenida. Por lo tanto hay que grabar uno ó cargar uno ya existente.

Para acceder a este módulo diríjase al Menú “PBX”, aparecerá por defecto la sección “Configuración PBX”, en el panel izquierdo escoja la opción “*System Recordings*” (Grabaciones del Sistema).



System Recordings

Add Recording

Step 1: Record or upload

If you wish to make and verify recordings from your phone, please enter your extension number here:

Alternatively, upload a recording in any supported asterisk format. Note that if you're using .wav, (eg, recorded with Microsoft Recorder) the file **must** be PCM Encoded, 16 Bits, at 8000Hz:

Step 2: Name

Name this Recording:

Click "SAVE" when you are satisfied with your recording

Grabaciones del sistema

La primera opción que tenemos es crear un anuncio grabándolo directamente, para esto ingresamos el número de extensión desde el cual queremos realizar la grabación, en este caso usaremos la extensión 201 y damos clic sobre el botón *Go*.

Luego de hacer esto, Asterisk estará esperando nuestra grabación en la extensión 201, para continuar marcamos *77, grabamos nuestro mensaje y finalmente presionamos la tecla # (numeral).

Para revisar nuestra grabación presionamos *99, ingresamos el nombre de nuestra grabación y damos clic sobre el botón “*Save*” (Guardar).

La segunda opción que tenemos es cargar una grabación creado en otro medio, para ello debemos tener un archivo soportado por Asterisk, damos clic sobre el botón “Examinar...”, buscamos nuestro archivo, luego procedemos a darle un nombre a esta grabación. Finalmente damos clic en “*Save*” (Guardar).

Configurar un IVR de bienvenida

El IVR nos permite grabar un mensaje de bienvenida y además podemos tener un menú controlado por teclado telefónico, a través de los 10 dígitos, y los símbolos # numeral y * asterisco. Con esto es posible enviar la llamada a otro destino o de nuevo al IVR que envió el anuncio.

Para acceder al módulo “IVR” diríjase al Menú “PBX”, aparecerá por defecto la sección “Configuración PBX”, en el panel izquierdo escoja la opción “IVR”.

Para grabar un mensaje de bienvenida diríjase a la sección “System Recordings” (Grabaciones del Sistema), por ejemplo:

IVR: “Gracias por llamar a Elastix, si usted conoce el número de extensión puede marcarlo ahora, caso contrario espere en la línea y un operador lo atenderá”.

Para ingresar un nuevo IVR NO es necesario completar todos los campos, pues para nuestro caso (un IVR de bienvenida), no necesitaremos opciones.

Los campos necesarios son los siguientes:

- **Change Name:** Cambiar el nombre, le pondremos Bienvenida.
- **Timeout:** Tiempo de espera (en segundos) antes de enrutar la llamada a un operador después de escuchar el mensaje de bienvenida. Para este ejemplo usaremos el número 3.
- **Enable direct dial:** Opción que permite a quien llama marcar una extensión directamente en caso de que la conozcan sin tener que esperar al operador.
- **Announcement:** Es el anuncio o mensaje de bienvenida que se grabó anteriormente. Aparecerá una lista con todos los mensajes disponibles.

Ahora procederemos a configurar ciertas opciones que son frecuentemente usadas, la primera es la opción 0 (cero) que nos permitirá ir directamente al operador y la segunda es también ir al operador pero una vez se ha escuchado todo el mensaje de bienvenida, más el tiempo de espera configurado anteriormente.

Digital Receptionist

Edit Menu Unnamed

Delete Digital Receptionist Unnamed

Change Name

Unnamed

Timeout

10

Enable Directory

☒

Directory Context

default

Enable Direct Dial

☒

Loop Before t-dest

☐

Loop Before i-dest

☐

Repeat Loops:

2

Announcement

None

Increase Options

Save

Decrease Options

Return to IVR

☐

☐ Announcements:

Feriadogye

☐ IVR:

Desde Elastix USA

☐ Terminate Call:

Hangup

☐ Extensions:

<1663> Jose Landivar

☐ Voicemail:

<1122> Edgar Landivar

☐ Queues:

Recepcion <2000>

☐ Misc Destinations:

prueba12

☐ Phonebook Directory:

Phonebook Directory

☐ Conferences:

Conferencia <1111>

Return to IVR

☐

☐ Announcements:

Feriadogye

☐ IVR:

Desde Elastix USA

☐ Terminate Call:

Hangup

☐ Extensions:

<1663> Jose Landivar

☐ Voicemail:

<1122> Edgar Landivar

☐ Queues:

Recepcion <2000>

☐ Misc Destinations:

prueba12

Formulario de creación de IVR

Entre las opciones del menú disponibles, en la parte izquierda existe un casillero donde se debe ingresar la opción. Para la primera opción (cero) pondremos este valor en el casillero, y asignamos alguna extensión configurada anteriormente, ésta extensión será el operador.

Estas extensiones aparecerán luego en la opción “Core”.

Ahora procedemos a configurar la segunda opción (permitir ir al operador luego de escuchar el mensaje de bienvenida más el tiempo de espera), para ello en el casillero de

la izquierda ingresamos la letra “t” lo que significa *timeout* y asignamos la extensión del operador.

Finalmente grabamos el IVR.

7.7 Salas de conferencias

Primero hay que aclarar que en Elastix se pueden administrar dos tipos de cuartos de conferencia: Los cuartos de conferencia permanentes y los cuartos de conferencia temporales. El primer tipo se puede administrar bajo el menú “PBX => PBX Configuration => Conferences”, mientras que el segundo se puede administrar bajo el menú “PBX => Conferencias”.

Los cuartos de conferencia no son otra cosa que una especie de extensión virtual en la que se puede alojar varias llamadas al mismo tiempo, dando con esto la ilusión de estar en una conferencia con muchas personas conversando al mismo tiempo.

Un cuarto de conferencia permanente es aquel que tiene un numero definitivo preasignado. Por ejemplo supongamos que queremos crear el cuarto 88999 y asignarlo al Gerente de Operaciones. De ahora en adelante las reuniones telefónicas de los lunes por las mañanas se pueden realizar simplemente marcando 88999 cuando llegue la hora pactada y todos los participantes saben que ese número de cuarto no cambiara de lunes a lunes.

Un cuarto de conferencia temporal se crea, como es obvio, para fines temporales. Por ejemplo la empresa va a participar de una exposición de productos en el extranjero y quiere agendar una serie de reuniones entre los proveedores y asesores externos para que se pongan de acuerdo en diferentes tópicos. Por lo tanto se decide asignar un cuarto de conferencia para cada asesor con el fin de que ellos mismos coordinen las conferencias que crean necesarias. Lógicamente no queremos que estos cuartos sean permanentes sino más bien que expiren luego de la feria ya que ellos no son trabajadores permanentes de la compañía.

Conference

New Conference

State: Past Conferences

Filter:

Show

Start

Previous (1 - 5 of 5)

Next

End

| Delete | Conference # | Conference Name | Starts | Ends | Participants |
|--------------------------|--------------|------------------------------|---------------------|---------------------|--------------|
| <input type="checkbox"/> | 73167 | Bruno Prueba | 2008-07-14 14:58:00 | 2008-07-14 15:58:00 | 10 |
| <input type="checkbox"/> | 91176 | prueba | 2008-07-21 11:56:00 | 2008-07-21 12:56:00 | 10 |
| <input type="checkbox"/> | 69298 | Prueba1 | 2008-08-05 10:12:00 | 2008-08-05 10:13:00 | 10 |
| <input type="checkbox"/> | 2316 | Elastix | 2008-09-04 16:43:00 | 2008-09-04 19:43:00 | 10 |
| <input type="checkbox"/> | 14079 | Prueba bruno | 2008-09-18 12:40:00 | 2008-09-18 13:40:00 | 10 |

Start

Previous (1 - 5 of 5)

Next

End

Listado de conferencias

En el presente apartado explicaremos cómo administrar conferencias temporales.

Al ingresar a conferencia podremos observar todas las conferencias concurrentes. Para ver las conferencias pasadas o futuras seleccione esta opción en “Estado” y de clic en “Mostrar”. Además se provee de un filtro usado para los nombres de las conferencias.

Para poder eliminar alguna o varias conferencias debemos seleccionarlasy dar clic sobre el botón Eliminar.

Si desea agregar una nueva conferencia de clic sobre el botón “Nueva Conferencia”.

Creación de nueva conferencia

Los campos requeridos para agregar una conferencia son:

- **Nombre de la Conferencia:** Necesario para identificar la conferencia
- **Número de la conferencia:** Se provee un número aleatorio para la conferencia usted puede cambiarlo si no está siendo usado por otra conferencia
- **Tiempo Inicio (PST/PDT):** Tiempo de inicio de la conferencia
- **Máximo Participantes:** No se podrán invitar más participantes una vez se completo el número máximo de participantes.

Para ver los detalles sobre una conferencia puede dar clic sobre el nombre de la conferencia.

Si desea administrar una conferencia concurrente, de clic sobre el enlace de participantes correspondiente a dicha conferencia.

Para invitar a alguien a la conferencia debe escogerlo de la lista proporcionada (lista de extensiones disponibles) y dar clic sobre el botón “Llamada a invitar”, deberá esperar hasta que se conteste la llamada.

Una vez haya invitado a alguien a la conferencia irán apareciendo en el reporte, si desea Silenciar a uno o varios invitados selecciónelos en la columna “Callado” y de clic sobre este botón (“Callado”).

Además puede botar de la conferencia a uno o varios invitados seleccionándolos en la columna “Botar” y de clic sobre este botón (“Botar”).

Si desea botar a todos los invitados de la conferencia de clic sobre “Botar a Todos”.

Si alguien a quien invitó no se muestra en la lista de invitados, o alguien a quien botó aun se muestra, de clic en el botón “Actualizar”. Este botón también actualizará el tiempo que lleva cada invitado en la conferencia.

Note que una vez de clic sobre el botón “Callado”, se actualizará el Estado del invitado, los estados posibles son: *Muted* (Callado), *UnMuted* (No Callado).

Para regresar a la interfaz de administración de todas las conferencias de clic en Cancelar.

7.8 Parqueo de llamadas

Un lote de llamadas es una extensión virtual con música en espera. Es decir que si se llama a esta extensión oiremos música en espera. A esta acción se le llama parquear una llamada.

Esta característica es muy útil cuando recibimos una llamada y no sabemos que hacer con ella. Por ejemplo, ingresa una llamada importante para el gerente de la empresa y el no esta en su puesto pero como es una llamada importante la operadora tiene que localizarlo así que mientras lo busca puede transferir al llamante en un lote de parqueo para que espere confortablemente con música en espera. Claro, muchos dirán que simplemente se lo pudo haber puesto *on hold* pero esto ocuparía la línea telefónica del teléfono de recepción impidiendo que nuevas llamadas ingresen. Una vez encontrado al esquivo gerente se le puede decir por ejemplo que tiene una llamada parqueada en el lote 3434. Luego, el mismo puede des-parquear dicha llamada y reiniciar la conversación.

Parking Lot Configuration

Parking Lot Options

Enable Parking Lot Feature ☐

Parking Lot Extension:

Number of Slots: 

Parking Timeout: 

Parking Lot Context:




Actions for Timed-Out Orphans

Parking Alert-Info:

CallerID Prepend:

Announcement: 

Destination for Orphaned Parked Calls:

- ☐ Announcements: 
- ☐ IVR: 
- ☐ Terminate Call: 
- ☐ Extensions: 
- ☐ Voicemail: 
- ☐ Queues: 
- ☐ Misc Destinations: 
- ☐ Phonebook Directory: 
- ☐ Conferences: 

Configuración de un nuevo lote de parqueos

Enable Parking Lot Feature:

Seleccionando esta opción con un visto se habilitarán la funcionalidad de parqueo de llamadas.

Parking Lot Extension:

Indica la extensión que servirá para parqueo de llamadas. Generalmente indica el primer *slot* de canales de parqueo.

Number Slots:

Indica el número de *slots* disponibles para parquear llamadas.

Parking Timeout:

El período de tiempo que una llamada puede permanecer parqueada. Si se alcanza este tiempo se intentará retornar la llamada al origen.

Parking Lot Context:

Este es el contexto para las llamadas parqueadas. Por defecto este contiene el contexto “parkedcalls” y no deberá ser cambiado.

Destination for Orphaned Parked Calls:

Indica el destino de una llamada cuando la llamada no puede ser parqueada y el llamador original tampoco puede ser alcanzado. Los destinos posibles pueden ser: Un anuncio, un IVR, Terminar la llamada, una extensión, una cola, una conferencia ó una aplicación personalizada.

7.9 Música en espera

En esta sección se puede configurar el “*Music On Hold*” que debería ser tocado, cuando se coloca una llamada en espera. Se pueden configurar varios archivos de audio y pueden ser usados en Colas.

Upload a .wav or .mp3 file:

Simplemente mediante el botón *browse* selecciono el archivo .mp3 ó .wav. Luego hago clic en “*upload*” y debería aparecer el archivo disponible en la lista de archivos de MOH que se muestran debajo de la opción de cargado de archivo.

On Hold Music

Category: default

Upload a .wav or .mp3 file:

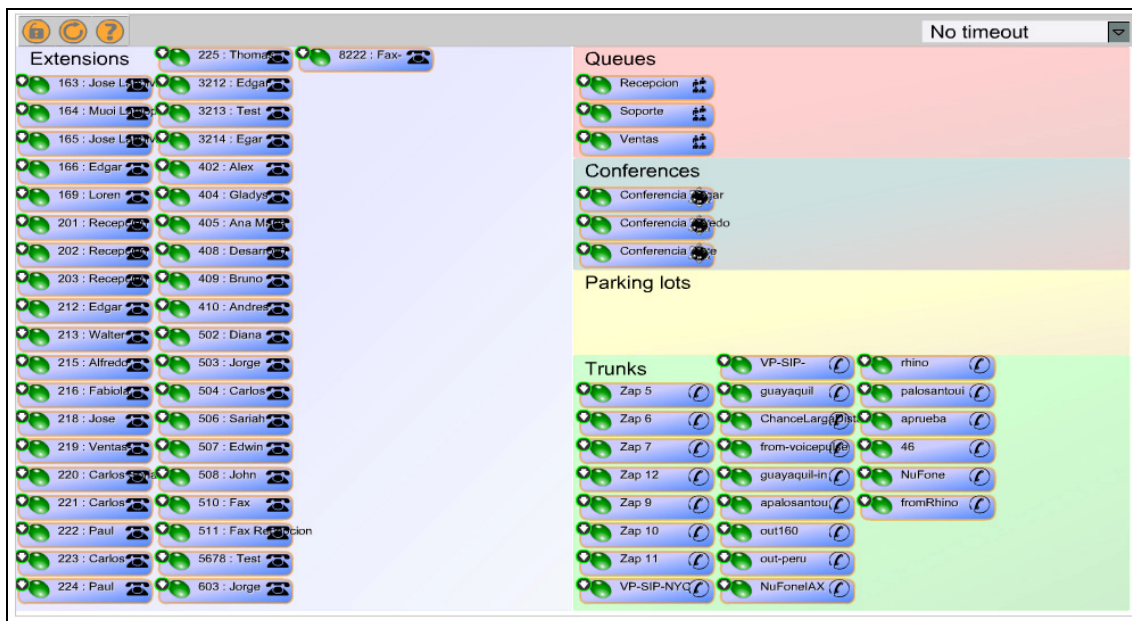
☒ Do not encode wav to mp3

| | |
|--------------------|--------|
| fpm-world-mix.wav | Delete |
| fpm-calm-river.wav | Delete |
| fpm-sunshine.wav | Delete |

Formulario de administración de música en espera

7.10 Flash Operator Panel (FOP)

El “Flash Operator Panel” del Menú “PBX” del Elastix es un manejador en flash de extensiones en Asterisk para monitorear los canales y terminales que se producen en un servidor con Asterisk. Es una herramienta muy útil para una recepcionista o para cualquiera que quiera tener una visión grafica del funcionamiento telefónico de la empresa.



Flash operador panel

7.11 Funciones de transferencia de llamadas

Por omisión Elastix viene configurado la funcionalidad de transferencia de llamada de la siguiente forma:

Transferencia ciega

Para trasferir una llamada de una manera ciega se debe digitar: ## seguido del número de extensión a donde se va a transferir la llamada.

Transferencia atendida

Para transferir una llamada de manera atendida se debe digitar: *2, seguido de la extensión.

Si desea cambiar estos “features” es necesario modificar el archivo `/etc/asterisk/features.conf`

8

Configuración de hardware

No siempre podemos agradar, pero siempre podemos tratar de ser agradables
-- Voltaire

A pesar de que Elastix puede funcionar sin necesidad de hardware telefónico, es necesario su uso si se requiere conexión a la PSTN. Su instalación y configuración se simplifica con Elastix gracias a que se dispone de una interfase Web especializada que facilita esta tarea.

En el presente capítulo nos referiremos básicamente a tarjetería PCI. Sin embargo pueden usarse otros dispositivos para conectarse a la PSTN como es el caso de los *gateways*.

8.1 Tipos de tarjetería y fabricantes

En este apartado nos referiremos básicamente a tarjetería telefónica PCI que es soportada por Elastix. Este *hardware* está disponible tanto para interfaces telefónicas analógicas como digitales y dentro de cada una de estas categorías existen básicamente las siguientes subcategorías:

- Analógico
 - FXO

- FXS
- Digital
 - E1/T1/J1
 - BRI

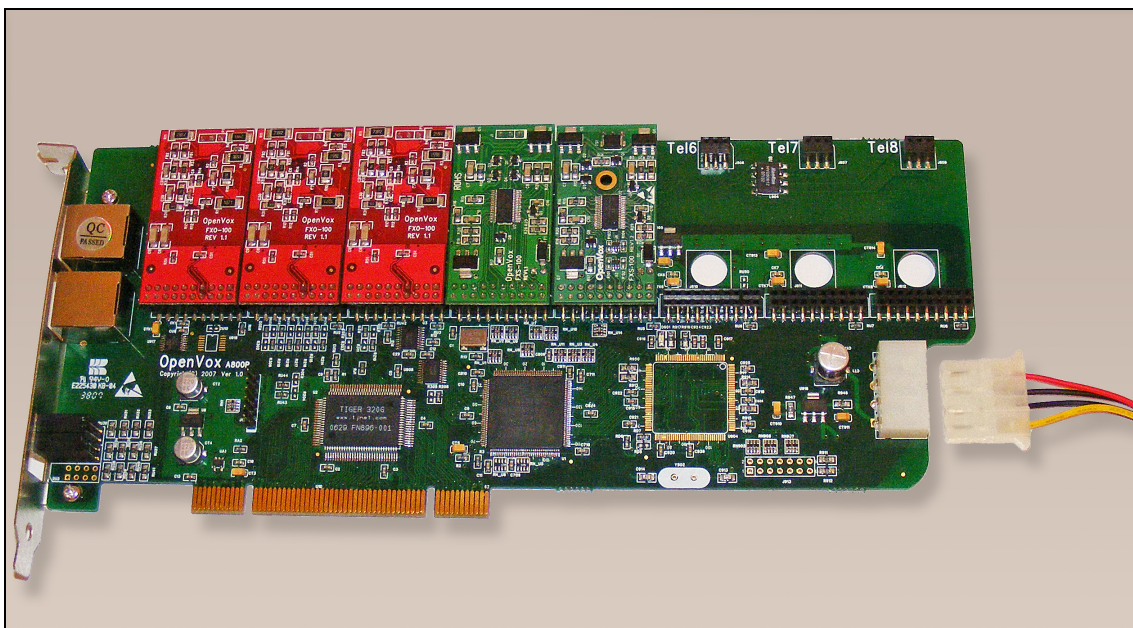
Elastix ha sido probado el soporte para muchos modelos y fabricantes de tarjetería. Los siguientes son los modelos soportados oficialmente a la fecha de redacción del presente libro:

- Digium
- OpenVox
- Sangoma
- Rhino
- Yeastar
- Xorcom (con interface USB)

8.2 Instalando físicamente la tarjetería

Esto no involucra ninguna tarea complicada, solo las precauciones obvias al instalar tarjetas electrónicas en un computador.

***Nota:** En caso de instalar hardware FXS es muy común que se requiera conectar un cable de poder a la tarjeta pues hay que recordar que las interfaces FXS deben proveer potencia eléctrica a los dispositivos que se conecten a ella.*

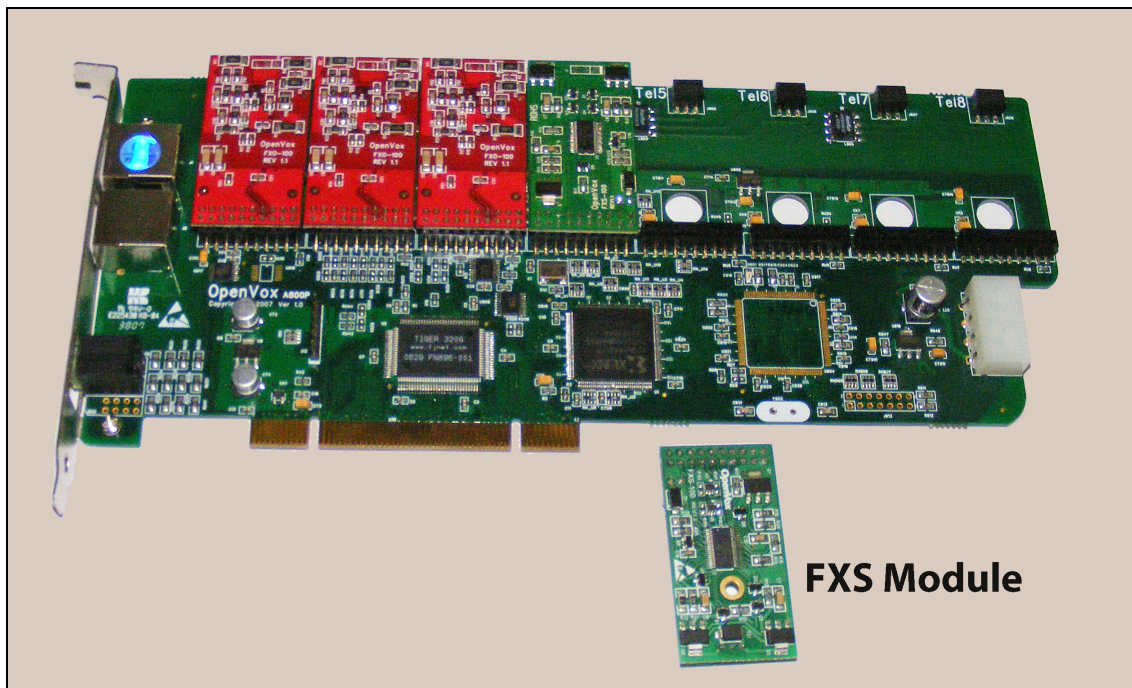


Tarjeta telefónica analógica de tipo modular con 3 puertos FXO y 2 FXS instalados

Luego encendemos el equipo y deberíamos verla enlistada a nivel de bus PCI

Hardware modular

Existen fabricantes que diseñan su hardware de manera modular, de tal forma que pueda ser expandible. Por ejemplo, demos un vistazo a esta tarjeta analógica marca OpenVox de 8 puertos poblada con módulos. Se puede observar que se ha extraído un módulo FXS de su base.



La misma tarjeta de la figura anterior a la cual se le ha extraído un módulo FXO

8.3 Detección de hardware

Una vez que la tarjetería ha sido instalada es hora de detectarla. En versiones muy antiguas de Elastix esta tarea se hacía manualmente ejecutando un comando desde la consola del equipo. Actualmente se cuenta con una interfase Web que detecta el hardware automáticamente y nos muestra gráficamente el hardware detectado.

Esta interfase se puede acceder desde el menú “System → Hardware Detector”. Para detectar nuevo hardware basta con presionar el botón “Detectar Nuevo Hardware”.

Hardware Detection

☐ Replace file zapata.conf

| PCI Slot # 1: WCTDM/0 "Wildcard TDM400P REV E/F Board 1" | | | |
|--|-----------------|-----------------|-----------------|
| Port # 1 FXO | Port # 2 FXO | Port # 3 FXO | Port # 4 FXO |
| Active | Active | Active | Active |

| PCI Slot # 2: WCTDM/0 "YSTDM8xx REV E Board 1" | | | | | | | |
|--|-----------------|-----------------|-----------------|-----------------|------------------|------------------|------------------|
| Port # 5 FXO | Port # 6 FXO | Port # 7 FXO | Port # 8 FXO | Port # 9 FXO | Port # 10 FXO | Port # 11 FXO | Port # 12 FXO |
| Active | Active | Active | Active | Active | Active | Active | Active |

Figura donde se observa 2 tarjetas detectadas

Esta detección no significa que nuestra tarjeta esté lista para funcionar ni mucho menos. Solo significa que los *drivers* han reconocido la tarjeta correctamente y que cierta configuración por defecto ha sido escrita para dicho hardware.

A partir de esta configuración por defecto el usuario puede modificar ciertos parámetros para que la tarjetería funcione cómo se desea. Estas configuraciones a bajo nivel se explicarán más adelante en este mismo capítulo.

***Nota:** Al momento de escribir este libro si se presiona el botón para detectar hardware TODAS las tarjetas serán re-detectadas y su configuración por defecto re-escrita. Si ya se había configurado una tarjeta previamente esta configuración será re-escrita. No se olvide de hacer los respaldos necesarios.*

8.4 Configuración de interfaces digitales a bajo nivel

La herramienta de detección de hardware basada en Web que viene con Elastix nos permite detectar el hardware de manera sencilla. Sin embargo, a pesar de que la herramienta puede configurar exitosamente el hardware digital en muchos casos, hay que tener en cuenta que la herramienta realiza su trabajo suponiendo muchos parámetros que pueden diferir de la realidad. Para corregir esto debemos realizar modificaciones a bajo nivel, es decir cambios en los archivos de configuración.

***Nota:** En este apartado nos referiremos básicamente a circuitos T-Carrier y E-Carrier (que es lo que mayormente se usa) basados en los drivers de Zaptel.*

Configuración de los *drivers* de Zaptel

Como ya habíamos visto en el archivo `zaptel.conf` se configura la tarjetería o hardware presente en el servidor; al menos los dispositivos que usan los *drivers* Zaptel. Esta parte es independiente de Asterisk. Es decir, bien podríamos utilizar el hardware y los *drivers* de Zaptel con otro software de PBX como Callweaver por ejemplo.

Para interfaces digitales en el archivo `zaptel.conf` se deben definir los *spans*. Un *span* es un grupo de canales digitales. Usualmente un *span* representa un puerto digital de la tarjeta. Por ejemplo, si la tarjeta contiene 4 puertos El entonces usualmente deberíamos definir 4 *spans*, uno por cada puerto.

Un *span* define en una sola línea, la cual tiene el siguiente formato.

```
span=(spannum) , (timing) , (LBO) , (framing) , (coding)
```

spannum

Es un identificador numérico único para etiquetar el span. Se debe empezar con el número 1 e ir incrementando este identificador secuencialmente de uno en uno.

Timing

Es un dígito que define la forma en la cual se manejará la sincronización del span. Los posibles valores son los siguientes.

| Valor | Significado |
|---------------|--|
| 0 | NO usar el <i>span</i> (telco) como fuente de sincronización. Zaptel proveerá la sincronización y la enviará al otro lado. Esto es más común cuando se conecta a través de este <i>span</i> una PBX antigua a la cual hay que proveerle sincronización. Normalmente si nos conectamos a una compañía telefónica no deberíamos tener ningún <i>span</i> con este valor de <i>timing</i> |
| 1 | Usar el <i>span</i> (telco) como fuente primaria de sincronización |
| 2 | Usar el <i>span</i> (telco) como fuente secundaria de sincronización |
| 3,4,5,.....,n | Usar el <i>span</i> (telco) como fuente de sincronización tercera, cuarta, quinta, etc... |

Nota: Solo el valor 0 puede ser repetido. Los demás valores del uno en adelante NO se deben repetir.

LBO (Line Build Out)

Es un dígito que representa la longitud del cable entre la tarjeta y el equipo (*modem*) provisto por la compañía telefónica. Casi siempre se puede dejar en 0 sin problemas. Sin embargo si tenemos un cable muy largo habrá que revisar la tablita siguiente.

| Valor | Longitud del cable o nivel de atenuación |
|-------|--|
| 0 | 0 dB / 0 - 133 pies |
| 1 | 133 - 266 pies |
| 2 | 266 - 399 pies |
| 3 | 399 - 533 pies |
| 4 | 533 - 655 pies |
| 5 | -7.5 dB |
| 6 | -15 dB |
| 7 | -22.5 dB |

Framing

Como recordaremos, los protocolos de comunicación digital *T-Carrier* y *E-Carrier* dividen la información en *frames*, donde cada *frame* transporta un número determinado de *timeslots*. Estos *frames* pueden ser de diferentes formatos y eso es lo que se define aquí.

Los valores posibles para T1 son **d4** o **esf**, mientras que para E1 son **cas** o **ccs**.

Coding

Es necesario codificar la información enviada para hacerla menos susceptible a errores. Esto es así porque la propia data se suele usar como fuente de sincronización (o reloj) y si se transmiten tramas largas de ceros (largos segmentos sin variación de voltaje) se puede perder dicha sincronización. Para evitar esto se codifica la data para evitar transmitir estas tramas largas de ceros. Por ejemplo la codificación **b8zs** significa *Bipolar with 8 Zeros Substitution*.

Los valores posibles para T1 son **ami** o **b8zs**, mientras que para E1 son **ami** o **hdb3**. En E1 también se puede necesitar añadir chequeo de errores **crc4** (Control de Redundancia Cíclica de 4 bits o *Cyclic Redundancy Check 4-bits*).

Al final reiniciamos Zaptel y ejecutamos el comando `lszaptel` para ver si nuestra tarjeta se encuentra bien configurada. Si todo está bien configurado y aún encontramos errores puede ser que el cable no esté conectado o algún error de parte del proveedor.

8.5 Configuración de troncales

Hasta este momento el hardware que hemos instalado no se encuentra relacionado con Asterisk de ninguna manera (a menos que se haya presionado el botón de sobrescribir `zapata.conf`).

zapata.conf

Asterisk cuenta con un módulo llamado `chan_zap` que sirve para conectarse con los *drivers* de Zaptel y de esta manera conectar Asterisk con hardware telefónico. Este módulo se configura a través de un archivo llamado `zapata.conf`

En el archivo `zapata.conf` básicamente se configura la señalización de los canales de voz y se mapean los canales de Asterisk con los circuitos o líneas telefónicas definidos en el archivo `zaptel.conf` que configuramos antes. Estos canales se pueden agrupar en grupos, también llamados troncales.

Es decir que si tengo 15 circuitos o líneas telefónicas que terminan en hardware telefónico de mi servidor los puedo controlar con Asterisk y troncalizarlas. Por ejemplo podría dividir esas líneas en 3 troncales debido a que mi cliente tiene 3 departamentos: ventas, soporte y financiero.

Veamos primero un archivo `zapata.conf` sencillo para ver los parámetros más importantes y después continuaremos explicando algunos de estos parámetros.

```
[trunkgroups]

[channels]
context=from-pstn
signalling=fxs_ls

; trunk #0
group=0
language=es
callprogress=no
context=from-zaptel
channel=1-5

; trunk #1
group=1
language=es
callprogress=no
context=from-zaptel
channel=6-10

; trunk #2
group=2
language=es
callprogress=no
context=from-zaptel
channel=11-15
```

Como vemos arriba hemos configurado tres troncales. Cada troncal comienza con la palabra `group` y se le asigna un identificador numérico. Por lo general se empieza con el número cero.

Dentro de cada troncal se pueden repetir los valores globales (es decir los que se encuentran en el encabezado, fuera de la definición de las troncales) y estos valores sobrescriben a los anteriores. Por ejemplo, se podría sobrescribir la señalización para una troncal específica.

Algunos parámetros importantes

signaling

Este parámetro define la señalización de cada canal o grupo de canales. Lo más común que veremos para circuitos analógicos será fxs_ls y para circuitos digitales pri_cpe. Sin embargo a continuación la tabla completa.

- em: E & M Immediate Start
- em_w: E & M Wink Start
- em_e1: E & M CAS signalling for E1 lines
- featd: Feature Group D (The fake, Adtran style, DTMF)
- featdmf_ta: Feature Group D (The real thing, MF (domestic, US)) through a Tandem Access point
- fgccama Feature Group C-CAMA (DP DNIS, MF ANI)
- fgccamamf Feature Group C-CAMA MF (MF DNIS, MF ANI)
- featdmf: Feature Group D (The real thing, MF (domestic, US))
- featb: Feature Group B (MF (domestic, US))
- fxs_ls: FXS (Loop Start)
- fxs_gs: FXS (Ground Start)
- fxs_ks: FXS (Kewl Start)
- fxo_ls: FXO (Loop Start)
- fxo_gs: FXO (Ground Start)
- fxo_ks: FXO (Kewl Start)
- pri_cpe: PRI signalling, CPE side
- pri_net: PRI signalling, Network side (for instance, side that provides the dialtone)
- sf: SF (Inband Tone) Signalling
- sf_w: SF Wink
- sf_featd: SF Feature Group D (The fake, Adtran style, DTMF)
- sf_featdmf: SF Feature Group D (The real thing, MF (domestic, US))
- sf_featb: SF Feature Group B (MF (domestic, US))
- e911: E911 (MF) style signalling. Originating switch goes off-hook, far-end winks, originating sends KP-911-ST, far-end gives answer supervision, Originating-end sends KP-0-ANI-ST
- The following are used for Radio interfaces:
- fxs_rx: Receive audio/COR on an FXS kewlstart interface (FXO at the channel bank)
- fxs_tx: Transmit audio/PTT on an FXS loopstart interface (FXO at the channel bank)
- fxo_rx: Receive audio/COR on an FXO loopstart interface (FXS at the channel bank)
- fxo_tx: Transmit audio/PTT on an FXO groundstart interface (FXS at the channel bank)

- `em_rx`: Receive audio/COR on an E&M interface (1-way)
- `em_tx`: Transmit audio/PTT on an E&M interface (1-way)
- `em_txrx`: Receive audio/COR AND Transmit audio/PTT on an E&M interface (2-way)
- `em_rtx`: same as `em_txrx` (for our dyslexic friends)
- `sf_rx`: Receive audio/COR on an SF interface (1-way)
- `sf_tx`: Transmit audio/PTT on an SF interface (1-way)
- `sf_txrx`: Receive audio/COR AND Transmit audio/PTT on an SF interface (2-way)
- `sf_rtx`: same as `sf_txrx` (for our dyslexic friends)

switchtype

Este parámetro es para circuitos digitales y especifica el tipo de switch PRI usado. Lo más común es `national` y `euroisdn`, dependiendo de en qué país nos encontremos claro está. Este dato lo debe proporcionar la compañía telefónica así que sintámonos en la libertad de preguntar. Los valores posibles son los siguientes.

- `national`: ISDN nacional tipo 2
- `ni1`: ISDN nacional tipo 1
- `dms100`: Nortel DMS100
- `4ess`: AT&T 4ESS
- `5ess`: Lucent 5ESS
- `euroisdn`: EuroISDN
- `qsig`: Protocolo usado en ISDN

8.6 Algunos cambios que se darán con el cambio de Zaptel a DAHDI

A partir de Elastix 1.5 el paquete Zaptel será reemplazado por el paquete DAHDI. Se ha tratado de que el proceso sea lo más transparente posible al usuario. Sin embargo, es muy importante tener en cuenta algunos detalles importantes y eso se cubrirá brevemente en el presente capítulo.

A pesar de que para algunos, el cambio de Zaptel a DAHDI es meramente un cambio de nombre, se ha aprovechado esta coyuntura para hacer algunos cambios en el código. Algunos hechos importantes a tener en cuenta son los siguientes.

- Zaptel ya no es mantenido por los desarrolladores de Asterisk y es en DAHDI donde se hacen los cambios. Por lo mismo, era vital migrar a DAHDI en Elastix.
- Los canceladores de eco pueden ser ahora configurados por canal en lugar de el enfoque anterior donde se debía seleccionar un solo cancelador de eco para toda la instalación
- Existen mejoras en el manejo de memoria con DAHDI
- Soporte para nuevo hardware

Algunos cambios de nombre a tener en cuenta

Si bien uno de los cambios más obvios de la migración de Zaptel a DAHDI es el cambio de nombre de los canales de ZAP a DAHDI; existen otros cambios de nombre que es muy importante tener en cuenta.

Cambios de nombre de módulos de kernel

| Módulo Zaptel | Módulo DAHDI |
|----------------|----------------------|
| zaptel.ko | dahdi.ko |
| ztd_eth.ko | dahdi_dynamic_eth.ko |
| ztd_loc.ko | dahdi_dynamic_loc.ko |
| ztdummy.ko | dahdi_dummy.ko |
| ztdynamic.ko | dahdi_dynamic.ko |
| zttranscode.ko | dahdi_transcode.ko |

Cambios de nombre de módulos de Asterisk

| Módulo Zaptel | Módulo DAHDI |
|-----------------|-------------------|
| chan_zap.so | chan_dahdi.so |
| app_zapbarge.so | app_dahdibarge.so |
| app_zapras.so | app_dahdiras.so |
| app_zapscan.so | app_dahdiscan.so |
| codec_zap.so | codec_dahdi.so |

Cambios de nombre de algunos comandos

| Comando Zaptel | Comando DAHDI |
|----------------|---------------|
| ztcfg | dahdi_cfg |
| ztmonitor | dahdi_monitor |

| | |
|-------------------------|---------------|
| ztscan | dahdi_scan |
| ztspeed | dahdi_speed |
| zttest | dahdi_test |
| zttool | dahdi_tool |
| zapconf o genzaptelconf | dahdi_genconf |

Cambios en archivos de configuración

| Archivo Zaptel | Archivo DAHDI |
|---------------------------|---|
| /etc/zaptel.conf | /etc/dahdi/system.conf |
| /etc/asterisk/zapata.conf | /etc/asterisk/chan_dahdi.conf |
| /etc/sysconfig/zaptel | /etc/dahdi/init.conf y /etc/dahdi/modules |

Para más información acerca de los cambios de Zaptel a DAHDI recomendamos el siguiente vínculo oficial:

<http://www.asterisk.org/zaptel-to-dahdi>

9

Configuración de teléfono IP

Es difícil despertar a aquel que simula estar dormido

-- Proverbio navajo

La configuración de un teléfono IP suele ser bastante sencilla. Como mínimo un teléfono IP necesita 3 valores para funcionar: El IP del servidor PBX (en este caso Elastix) el usuario (comúnmente la extensión) y la contraseña de dicho usuario.

Aquí mostraremos al detalle cómo configurar un teléfono IP físico y otro basado en software ya que los *softphones* se han vuelto bastante populares en los últimos tiempos. En estos teléfonos se usará tecnología SIP para su configuración puesto que es la tecnología más popular en la actualidad, pero sintámonos confiados de configurar un teléfono en tecnología IAX. La configuración no diferirá mucho de lo aquí mostrado. En general para la mayoría de los casos usaremos SIP en nuestros ejemplos a menos que estemos hablando de teléfonos que se conectan remotamente ya que en estos casos será preferible usar IAX debido a los problemas de NAT que sufre el protocolo SIP.

Finalmente explicaremos un método bastante útil si nos encontramos en el caso de configurar un número grande de extensiones. Este método nos permite pre-configurar los teléfonos desde una herramienta llamada “*endpoint configurator*” disponible en Elastix.

9.1 Configuración de *softphone*

Los teléfonos en software o *softphones* se han vuelto muy populares últimamente por el hecho del ahorro que puede significar su uso al no tener que comprar teléfonos físicos. El ahorro es mayor mientras más extensiones se tengan y es por eso que esta solución es muy popular en *call centers*.

Por supuesto antes de empezar a configurar el *softphone* habrá que crear una extensión desde Elastix. Ya hemos explicado esto anteriormente en este libro así que omitiremos los detalles. En este ejemplo se supondrá la extensión 5678 con tecnología SIP y secret 5678.

El *softphone* que vamos a configurar será el Zoiper que está disponible en versión gratuita y comercial. La versión gratuita es suficiente para que podamos hablar cómodamente de un punto a otro así que basaremos nuestro ejemplo en esta versión.

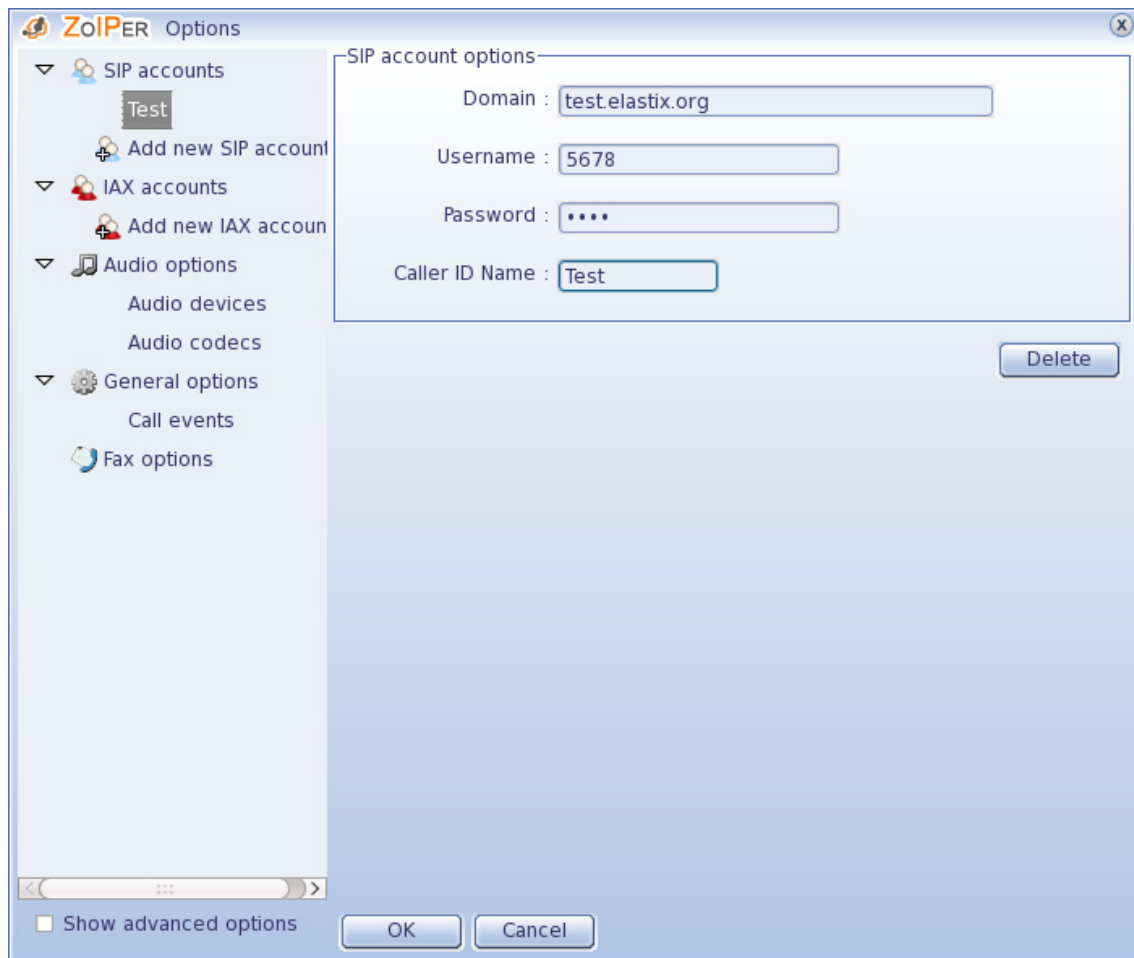
Para descargar el Zoiper debemos visitar el sitio <http://www.zoiper.com>

Suponemos que el lector tiene la habilidad suficiente para instalar el software sin problemas. Una vez instalado lo ejecutamos y aparecerá una ventana como la siguiente.



Pantalla inicial del software ZoIPer

A continuación presionamos el botón con la figura de la herramienta y accederemos al menú de opciones. Allí veremos una pantalla con opciones para configurar nuevas extensiones SIP y también IAX, pues el Zoiper cuenta con soporte para ambos protocolos. En nuestro ejemplo configuraremos una extensión SIP así que veremos una pantalla como la siguiente.

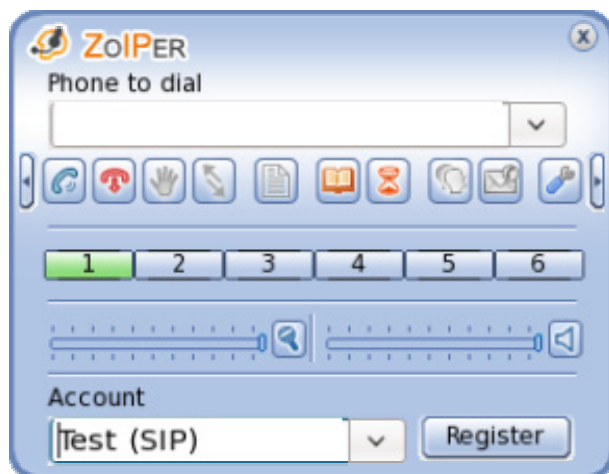


Configurando una cuenta SIP

Llenaremos dicha pantalla con los siguientes datos:

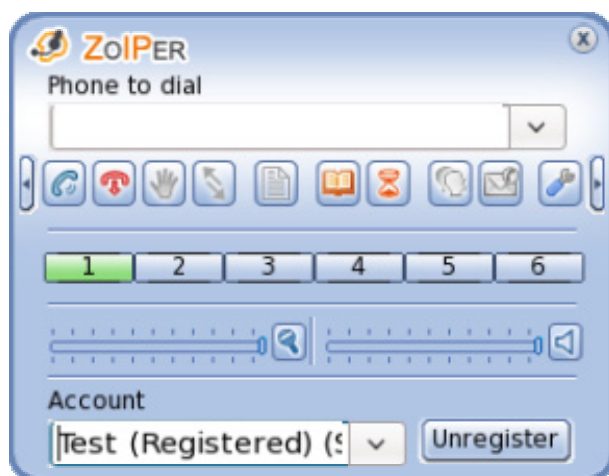
- **Domain:** La dirección IP o nombre de dominio del servidor Elastix.
- **Username:** 5678
- **Password:** 5678
- **Caller ID Name:** El nombre que aparecerá en los teléfonos que reciban nuestras llamadas

A continuación volvemos a la pantalla inicial del Zoiper y allí debemos registrarnos con la extensión que creamos. Para esto debemos seleccionar la extensión creada en el combo de la parte inferior y dar clic en el botón “Register”.



Antes de registrar la cuenta SIP

Si el Zoiper se registra correctamente veremos que junto al nombre de nuestra cuenta “Test” aparecerá la palabra “Registered”.



Luego de que la cuenta SIP ha sido registrada

Listo, ahora voy a llamar desde otro teléfono para ver si se puede alcanzar a mi nueva extensión 5678 y podemos mantener una conversación.



Llamada ingresando a nuestro *softphone*

Todo bien, la llamada fue recibida y la comunicación transcurrió sin inconvenientes.

***Nota:** En ocasiones nos encontraremos solos configurando un Elastix para un cliente y se nos complica un poco hacer pruebas con otro teléfono. En estos casos puede ser útil probar la comunicación llamando a la extensión de ECO. Esta extensión contestará automáticamente y reproducirá el mismo audio que enviamos.*

9.2 Configuración de teléfono físico

Para configurar un teléfono IP debemos tener en cuenta dos aspectos:

1. Configuración de red
2. Registro del dispositivo con el servidor Elastix

Para ilustrar de mejor manera los puntos mencionados, utilizaremos un teléfono Polycom SoundPoint IP 330.



El teléfono Polycom 330 que configuraremos

Configuración de red

La configuración de red puede realizarse considerando que el teléfono obtendrá su IP mediante DHCP (lo más común) o que se le asignará una dirección IP estática. Explicaremos los dos métodos. Para el primer caso explicaremos cómo obtener el IP del teléfono más que cómo configurarla ya que se asigna automáticamente. En el caso de IP estática explicaremos cómo asignarle un IP al teléfono.

Al final del proceso conoceremos la dirección IP del dispositivo y con este dato podremos acceder a la interfase de configuración Web que nos ofrece este modelo de teléfono.

DHCP

Mediante esta opción, el teléfono obtendrá una dirección IP de manera automática. No se debe olvidar de que para que éste lo pueda hacer, dentro de la red deberá haber un

servidor DHCP. Elastix viene con un servidor DHCP embebido que se puede activar fácilmente desde la interfase Web.

Primero necesitamos conocer la dirección IP que adquirió el teléfono. A continuación se detalla el procedimiento para determinarla.

Presionar "Menu -> Status -> Network -> TCP/IP Parameters"

Aquí aparecerá la dirección IP. Anotémosla para luego poder acceder al teléfono desde la interfase Web.

IP estática

Esta opción le será útil si desea contar con un registro exacto de las direcciones IP de sus teléfonos y/o si desea colocar sus dispositivos IP en segmentos específicos de su red. En esta opción, necesitará ingresar la siguiente información.

Dirección IP

Recuerde que la IP que le vaya a asignar a este equipo no la debe tener ningún otro equipo, debe ser única en su red. Vamos a suponer que el teléfono estaba previamente configurado con DHCP ya que así viene de fábrica.

Presionar "Menu -> Settings -> Advanced -> (ingresar clave. Por omisión es 456) -> Admin Settings -> Network Configuration -> DHCP Client -> Editar -> Disabled -> Phone IP Address"

Aquí procederemos a ingresar la dirección IP.

Máscara de red

Si usted no ingresa una máscara de red adecuada, su teléfono IP no podrá registrarse con su central Elastix. Esos problemas ocurren con mayor frecuencia cuando los teléfonos IP están dentro de diferentes segmentos de redes que la central Elastix, es ahí donde se debe seleccionar la máscara adecuada.

Seleccionar "Subnet Mask -> (Ingresar la máscara de red)

Puerta de enlace

En el caso de que su teléfono se encuentre en otro segmento de red, deberá ingresar la puerta de enlace adecuada para que su teléfono pueda ver a su Elastix.

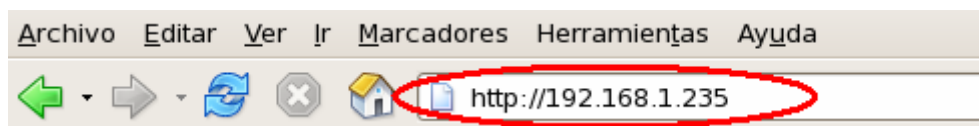
Seleccionar "IP Gateway -> (Ingresar la IP de la puerta de enlace)

Luego de cualquier cambio debemos seleccionar "Exit" y luego "Save Config". Debemos esperar a que el teléfono termine de reiniciarse

Registro del dispositivo con el servidor Elastix

Esta configuración se la pueda realizar desde el mismo teléfono o desde un navegador Web. En este caso utilizaremos el navegador: Mozilla Firefox.

Como ya conocemos la dirección IP de nuestro teléfono la accedemos desde el navegador Web. En el siguiente ejemplo el IP de nuestro teléfono es la 192.168.1.235

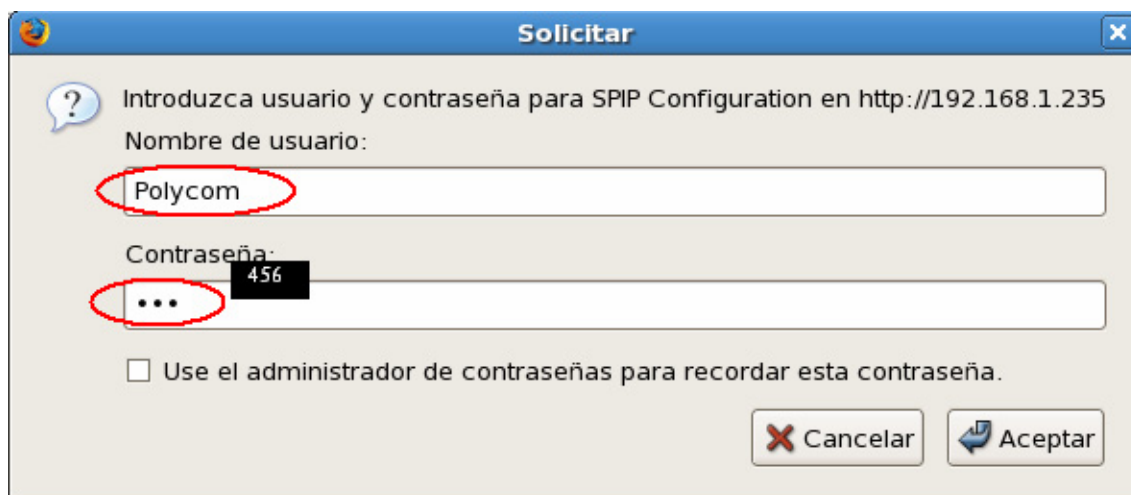


Barra del navegador Firefox

IP del servidor Elastix

Para comenzar debemos seleccionar la pestaña que dice "SIP".

La interfaz Web nos pedirá el usuario y la clave del teléfono para poderlo configurar. El usuario y clave de fábrica son "Polycom" y "456" respectivamente.



Diálogo de autenticación vía usuario y contraseña

Aquí podremos configurar algunos parámetros SIP como el servidor SIP, es decir nuestra central Elastix, así como también el “Outbound Proxy” que viene a ser lo mismo en nuestro caso. La configuración adecuada se muestra en la siguiente figura, donde se supone que la dirección IP del servidor Elastix es la 192.168.1.66

POLYCOM SoundPoint IP Configuration

Home General Network **SIP** Lines

SIP Configuration Parameters:

Servers Local Settings

Servers

Outbound Proxy

Address 192.168.1.160

Port 5060

Transport DNSnaptr

Server 1

Address 192.168.1.160

Port 5060

Transport DNSnaptr

Expires

Register 1

Retry Time Out 0

Retry Max Count 0

Line Seize Time Out 30

Server 2

Address

Port

Transport DNSnaptr

Expires

Register

Retrv Time Out

Puerto de registro para la extensión

dirección IP del Elastix

Interfase Web de configuración de parámetros SIP del teléfono

Debemos tomar en cuenta de que la dirección de la Elastix puede estar dentro de nuestro segmento de red, dentro de otro segmento de red, o incluso una IP fuera de nuestra red (IP pública externa).

En esta pantalla también podemos configurar el puerto SIP donde se registrará nuestro teléfono. Por omisión en Elastix este valor es 5060.

Extensión

Seleccione la opción "Lines" y configure aquí su extensión como se muestra en la siguiente figura. El parámetro "Auth Password" debe ser igual al parámetro "secret" que se ingresó cuando se creó la extensión en la interfase Web del Elastix.

The screenshot shows the Polycom SoundPoint IP Configuration web interface. The top navigation bar includes the Polycom logo and the title 'SoundPoint IP Configuration'. Below this is a menu with tabs: Home, General, Network, SIP, and Lines. The 'Lines' tab is selected, and the 'Line Parameters' section is visible. Under 'Line Parameters', there are three sub-sections: Line 1, Line 2, and Line 3. The 'Line 1' section is expanded, showing a form for configuring the first line. The form is divided into two main sections: 'Identification' and 'Server 1'. In the 'Identification' section, the 'Display Name' is 'Jorge Gutierrez', the 'Address' is '503', the 'Auth User ID' is '503', and the 'Auth Password' is masked with dots. The 'Label' is '503', the 'Type' is 'Private', and the 'Num Line Keys' is '1'. In the 'Server 1' section, the 'Address' is empty, the 'Port' is '5060', the 'Transport' is 'DNSnaptr', and the 'Line Seize Time Out' is '30'. Red circles highlight the 'Address', 'Auth User ID', and 'Auth Password' fields in the 'Identification' section.

| Line Parameters: | | |
|-----------------------|---|--------|
| Line 1 | Line 2 | Line 3 |
| Line 1 | | |
| Identification | | |
| Display Name | Jorge Gutierrez | |
| extensión / teléfono | Address | 503 |
| | Auth User ID | 503 |
| clave / autenticación | Auth Password | •••• |
| Label | 503 | |
| Type | <input checked="" type="radio"/> Private <input type="radio"/> Shared | |
| Third Party Name | | |
| Num Line Keys | 1 | |
| Calls Per Line | | |
| Server 1 | | |
| Address | | |
| Port | 5060 | |
| Transport | DNSnaptr | |
| Expires | | |
| Register | 1 | |
| Retry Time Out | | |
| Retry Max Count | | |
| Line Seize Time Out | 30 | |

Configuración de la línea SIP

Configuración Opcional

En esta sección veremos cómo configurar el Polycom para que muestre la fecha y hora del Elastix.

En el menú debemos dar clic en la pestaña "General" y en el campo "SNTP Server" (bajo la sección time) ingresar la dirección IP del servidor Elastix.

Adicionalmente debemos seleccionar el valor adecuado de su zona horaria en "GMT Offset"

POLYCOM SoundPoint IP Configuration

Home **General** Network SIP Lines

General Configuration Parameters:

| | | |
|------------------|--------------|------------------|
| User Preferences | Time | Audio Processing |
| Sampled Audio | Microbrowser | Logging |

User Preferences

| | |
|---------------------------|---|
| Headset Memory Mode | <input type="radio"/> Enabled <input checked="" type="radio"/> Disabled |
| Use Directory Names | <input type="radio"/> Enabled <input checked="" type="radio"/> Disabled |
| One Touch Voice Mail | <input type="radio"/> Enabled <input checked="" type="radio"/> Disabled |
| Bypass Instant Message | <input checked="" type="radio"/> Enabled <input type="radio"/> Disabled |
| Welcome Sound (All Boot) | <input checked="" type="radio"/> Enabled <input type="radio"/> Disabled |
| Welcome Sound (Warm Boot) | <input type="radio"/> Enabled <input checked="" type="radio"/> Disabled |
| top | Submit |

Time

Synchronization

| | | |
|--------------------|---------------|-----------------------------------|
| SNTP Server | 192.168.1.160 | Dirección IP del servidor Elastix |
| GMT Offset | -5 | Uso horario |
| SNTP Resync Period | 86400 | |

Daylight Savings

| | |
|------------------|---|
| Daylight Savings | <input checked="" type="radio"/> Enabled <input type="radio"/> Disabled |
|------------------|---|

Configuración del servidor de tiempo

Nota: Si al querer digitar alguna de sus extensiones su teléfono marca inmediatamente una parte de esta y no espera a que termine de ingresar todos los dígitos de la extensión, revise la opción "Digitmap", dentro de la sección "Local Settings" dentro del menú "SIP", o en su defecto elimínela por completo.

| Local Settings | |
|--|---|
| Local SIP Port | <input type="text"/> |
| Calls Per Line Key | <input type="text"/> |
| New SDP Type | <input type="radio"/> Enabled <input checked="" type="radio"/> Disabled |
| LCS Support | <input type="radio"/> Enabled <input checked="" type="radio"/> Disabled |
| Non Standard Line Seize | <input checked="" type="radio"/> Enabled <input type="radio"/> Disabled |
| Digitmap | <div style="border: 2px solid red; border-radius: 50%; padding: 5px;"> <input type="text" value="[2-9] 11 0T 01xxxx.T"/> <input type="text" value="[0-1] [2-9] xxxxxxxxxx "/> <input type="text" value="[2-9] xxxxxxxxxx "/> </div> |
| Digitmap Timeout | <input type="text" value="3 3 3 3 3"/> |
| Remove End-Of-Dial Marker | <input checked="" type="radio"/> Enabled <input type="radio"/> Disabled |
| Digitmap Impossible Match | <input type="text" value="0"/> |
| <div style="display: flex; justify-content: space-between;"> top Submit </div> | |

Cambiando el mapa de dígitos

9.3 Utilizando el *endpoint configurator*

En ocasiones la instalación de teléfonos IP se complica cuando son muchos y más aún cuando la configuración no es tan trivial como en el *softphone* que acabamos de configurar. Existen por ejemplo algunos modelos de teléfonos que se pueden configurar desde una interfase Web embebida en el propio teléfono e inclusive existen otros que se tienen que configurar a través de una sesión telnet.

La distro Elastix está preparada para estos y cuenta con una interfase que nos permitirá configurar muchos teléfonos de una manera más sencilla. Esta herramienta recibe el nombre de *Endpoint Configurator*.

El módulo *Endpoint Configurator* como primera acción realiza un rastreo (*scan*) de la red o subred que se le suministre. Algo de notar es que cuando se ingresa al módulo, la red o subred por omisión es aquella en la que se encuentra la propia central Elastix; el *scan* que se hace a dicho segmento de red obtiene internamente información de MAC e IP de los dispositivos de red conectados.

Endpoint Configuration

Endpoint Scan

192.168.1.0/24

Nota: La dirección MAC también le sirve al endpoint configurator como identificador único del teléfono. Si se usara la dirección IP ésta podría cambiar con el tiempo, pero con la dirección MAC no ocurre esto.

Luego de obtener la información de todos los *endpoints* de algún segmento de red, el módulo de forma inteligente trata de identificar los equipos que pudieran ser teléfonos IP y también trata de recabar información de fabricante y modelo para elegir cuáles puede configurar basándose en su base de conocimiento de modelos de teléfonos.

Cuando ya se tiene el grupo de teléfonos a configurar se presenta la información en la Web interfase de Elastix, se mostrara algo como se ilustra en la figura siguiente.

Endpoint Configuration

Endpoint Scan192.168.1.0/24

Start / Previous (1 - 19 of 19) Next End

| Set | Unset | MAC Address | IP Address | Vendor | Phone Type | User Extension | Status |
|--------------------------|--------------------------|-------------------|-------------------------------|----------------------------------|------------------|------------------|---------|
| <input type="checkbox"/> | <input type="checkbox"/> | 00:09:45:54:A7:A0 | 192.168.1.82 | Atcom / Palmmicro Communications | -- Unselected -- | -- Unselected -- | Not Set |
| <input type="checkbox"/> | <input type="checkbox"/> | 00:09:45:52:CE:AB | 192.168.1.84 | Atcom / Palmmicro Communications | -- Unselected -- | -- Unselected -- | Not Set |
| <input type="checkbox"/> | <input type="checkbox"/> | 00:09:45:53:1B:45 | 192.168.1.134 | Atcom / Palmmicro Communications | -- Unselected -- | -- Unselected -- | Not Set |
| <input type="checkbox"/> | <input type="checkbox"/> | 00:04:F2:19:CE:6D | 192.168.1.137 | Polycom / Polycom | -- Unselected -- | -- Unselected -- | Not Set |
| <input type="checkbox"/> | <input type="checkbox"/> | 00:04:F2:12:FE:F6 | 192.168.1.177 | Polycom / Polycom | -- Unselected -- | -- Unselected -- | Not Set |
| <input type="checkbox"/> | <input type="checkbox"/> | 00:04:F2:19:CB:F9 | 192.168.1.220 | Polycom / Polycom | -- Unselected -- | -- Unselected -- | Not Set |
| <input type="checkbox"/> | <input type="checkbox"/> | 00:09:45:53:1B:3B | 192.168.1.222 | Atcom / Palmmicro Communications | -- Unselected -- | -- Unselected -- | Not Set |
| <input type="checkbox"/> | <input type="checkbox"/> | 00:0E:08:DA:89:96 | 192.168.1.224 | Linksys / Sipura Technology | -- Unselected -- | -- Unselected -- | Not Set |
| <input type="checkbox"/> | <input type="checkbox"/> | 00:04:F2:19:CB:D8 | 192.168.1.225 | Polycom / Polycom | -- Unselected -- | -- Unselected -- | Not Set |
| <input type="checkbox"/> | <input type="checkbox"/> | 00:09:45:52:28:6A | 192.168.1.226 | Atcom / Palmmicro Communications | -- Unselected -- | -- Unselected -- | Not Set |
| <input type="checkbox"/> | <input type="checkbox"/> | 00:09:45:52:53:DF | 192.168.1.227 | Atcom / Palmmicro Communications | -- Unselected -- | -- Unselected -- | Not Set |

Start / Previous (1 - 19 of 19) Next End

Listado inicial de teléfonos encontrados en la red. Sin configurar aún.

Es una lista de todos los teléfonos que se han encontrado en el segmento de red dado y que el Elastix conoce o podrá configurar según su base de datos. Hay que recalcar que actualmente Elastix en este módulo solo tiene soporte garantizado para Atcom, Polycom y Linksys. Se prevee para versiones futuras de Elastix tener soporte para Cisco, Aastra, Grandstream y Snom.

Como podrá notar el reporte muestra las columnas *MAC Address*, *IP Address* y *Vendor* que nos ayudará a reconocer de cual teléfono IP se trata y con esto podremos relacionar mentalmente su ubicación física.

| MAC Address | IP Address | Vendor |
|-------------------|-------------------------------|----------------------------------|
| 00:09:45:54:A7:A0 | 192.168.1.82 | Atcom / Palmmicro Communications |
| 00:09:45:52:CE:AB | 192.168.1.84 | Atcom / Palmmicro Communications |
| 00:09:45:53:1B:45 | 192.168.1.134 | Atcom / Palmmicro Communications |
| 00:04:F2:19:CE:6D | 192.168.1.137 | Polycom / Polycom |
| 00:04:F2:12:FE:F6 | 192.168.1.177 | Polycom / Polycom |
| 00:0E:08:DA:89:96 | 192.168.1.224 | Linksys / Sipura Technology |

Vista acercada de campos MAC, dirección IP y Fabricante

Luego hay dos columnas *Phone type* y *User extension* para una configuración rápida de la extensión para el teléfono. En *Phone type* podrá elegir el modelo del teléfono y solo se listaran aquellos que Elastix sabe configurar y reconocer. Tener en cuenta que para configurar los teléfonos se deben haber creados previamente extensiones desde el menú *PBX->PBX Configuration*.

Otra limitante es que las extensiones deben ser SIP, no están soportadas extensiones de ningún otro protocolo.

A continuación se listaran modelos y fabricantes, lo que esta de color amarillo son aquellos que seran soportados en proximas versiones Elastix:

| Polycom | Atcom | Linksys | Aastra | Cisco | Grandstream | Snom |
|--------------|--------|---------|---------|-------|-------------|------|
| IP 301 | AT 320 | SPA921 | 480i | 7960 | HT386 | 360 |
| IP 330/320 | AT 530 | SPA922 | 480i CT | 7940 | GXP2000 | 320 |
| IP 430 | | SPA941 | 9133i | 7970 | | |
| IP 501 | | SPA942 | 53i | 7971 | | |
| IP 601 | | SPA962 | 55i | | | |
| IP 550 | | SPA841 | 57i | | | |
| IP 650 | | | 57i CT | | | |
| SoundStation | | | | | | |
| IP 4000 | | | | | | |

Configuración de un teléfono ATCOM, modelo AT-530

A continuación explicaremos paso a paso cómo configurar un teléfono marca ATCOM, modelo AT-530. Este es un modelo de teléfono de bajo costo que soporta tanto SIP como IAX.

Para configurarlo primero elegimos el modelo y la extensión a configurar del listado mostrado en el *Endpoint Configurator*. Cuando un teléfono es configurado por primera vez mediante este módulo, éste presenta un estatus “*Not Set*” lo que significa que todavía la central Elastix no tiene configuración acerca del él.

| MAC Address | IP Address | Vendor | Phone Type | User Extension | Status |
|-------------------|-------------------------------|----------------------------------|------------|--------------------------|---------|
| 00:09:45:53:1B:45 | 192.168.1.134 | Atcom / Palmmicro Communications | AT 530 | Desarrollo Elastix <408> | Not Set |

Línea correspondiente al teléfono AT-530 detectado en la red

En la parte izquierda vamos a elegir la casilla, y luego presionamos el botón “*Set*”. Inmediatamente aparece la ventana de confirmación en la cual si se está seguro se acepta y el proceso de configuración empezará. Esto puede tomar algo de tiempo dependiendo del número de teléfonos que haya elegido, en este caso por ser un ejemplo hay solo un teléfono a configurar.

| <input type="checkbox"/> | <input type="button" value="Set"/> | <input type="button" value="Unset"/> | MAC Address | IP Address | Vendor |
|-------------------------------------|------------------------------------|--------------------------------------|-------------------|-------------------------------|----------------------------------|
| <input checked="" type="checkbox"/> | | | 00:09:45:53:1B:45 | 192.168.1.134 | Atcom / Palmmicro Communications |

Vista ampliada con los datos de nuestro teléfono AT-530

Luego del proceso vamos a ver que el reporte se actualizó y el estatus del teléfono es “*Configured without incident*”. Esto indica que la central Elastix hizo varias cosas.

- Primero creó el (o los) archivos de configuración necesarios en la ruta /tftpboot/ (ruta en la central Elastix) y en caso de este ejemplo solo se creó el archivo `atc000945531b45.cfg` donde 000945531b45 representa la dirección MAC del teléfono.
- Segundo creó un registro en la base de datos de Elastix para mantener un historial del teléfono configurado.
- Tercero, mediante conexión *telnet* (solo para los ATCOM), realizó acciones de cargar el archivo de configuración al teléfono, guardar al archivo y reiniciar. Por ello usted podrá notar que el teléfono se va a reiniciar.

| IP Address | Vendor | Phone Type | User Extension | Status |
|-------------------------------|----------------------------------|------------|--------------------------|------------------------------|
| 192.168.1.134 | Atcom / Palmmicro Communications | AT 530 | Desarrollo Elastix <408> | Configured without incident. |

Teléfono AT-530 configurado

Hay que tener ciertas consideraciones en cuanto a la conexión *telnet* en los teléfonos ATCOM y es que para poder establecerse es necesario de un usuario y contraseña que por efectos de simplicidad en Elastix se usan los que vienen por omisión de fábrica. Para el caso del modelo AT-530 son usuario *admin* con clave *admin* y para el caso del AT-320 la clave es 12345678.

Regresando a interpretar los Status del teléfono tenemos.

| Estatus | Color | Descripción |
|--|---------|--|
| <i>Not Set</i> | Naranja | configurado aun por la central Elastix |
| <i>Configured without incident</i> | Verde | La central configuró el teléfono sin problemas |
| <i>UPDATE: Secrets no equals in FreePBX and Endpoint</i> | Rojo | Es necesario actualizar. La razón es porque ha cambiado la clave en la central (FreePBX) y en la base de endpoint.db aun se mantiene la anterior |

Si no hay incidentes entonces ahora nuestro teléfono IP ya esta configurado desde la central Elastix y cuando se desee cambiar de extensión sólo vamos al *endpoint configurator*, hacemos un par de clics y listo.

Nota: Existe el botón “Unset”. La acción que ejecuta éste es sólo a nivel del módulo. La central Elastix borra el registro histórico relacionado con el teléfono IP de la base de su base de datos interna y borra los archivos de configuración relacionados en la ruta */tftpboot/*. Esto indica que el teléfono en si mantiene la configuración que tuvo.

Configuración de un teléfono Polycom SoundPoint IP 501

A continuación mostraremos un ejemplo de cómo configurar un teléfono Polycom SoundPoint IP501.

Primero elegimos el modelo y la extensión a configurar. Como en el caso del teléfono ATCOM podemos partir de que es la primera vez que se lo va a configurar desde la central Elastix y por ende notamos el Status “*Not Set*”.

| MAC Address | IP Address | Vendor | Phone Type | User Extension | Status |
|-------------------|-------------------------------|-------------------|------------|--------------------------|---------|
| 00:04:F2:13:14:70 | 192.168.1.235 | Polycom / Polycom | IP 501 | Desarrollo Elastix <408> | Not Set |

Línea correspondiente al teléfono Polycom detectado en la red

En la parte izquierda vamos a elegir la casilla, y luego presionamos el botón “Set”. Inmediatamente aparece la ventana de confirmación en la cual si esta seguro se acepta y el proceso de configuración empezará. Esto puede tomar algo de tiempo dependiendo del número de teléfonos que haya elegido, en este caso por tratarse de un ejemplo hay solo un teléfono a configurar.

| | | | | |
|-------------------------------------|--------------------------------------|--------------------|-------------------------------|-------------------|
| <input type="button" value="Set"/> | <input type="button" value="Unset"/> | MAC Address | IP Address | Vendor |
| <input checked="" type="checkbox"/> | | 00:04:F2:13:14:70 | 192.168.1.235 | Polycom / Polycom |

Vista ampliada con los datos de nuestro teléfono Polycom IP501

Luego del proceso vamos a ver que el reporte se actualizó y el estatus del teléfono es “Configured without incident”, esto indica que la central Elastix hizo varias cosas.

Primero creo los archivos de configuración necesarios en la ruta /tftpboot/ (ruta en la central Elastix). Estos archivos para el caso de los teléfonos Polycom son algunos.

| Archivo | Descripción |
|---------------------|--|
| 0004f2131470.cfg | El 0004f2131470 representa la dirección MAC y este archivo es un encabezado que le indica al teléfono lo que se va a hacer durante el proceso de actualización y configuración |
| 0004f2131470reg.cfg | Contiene información de la (o las) extensiones que va ha poseer el teléfono |
| polycom | Realmente es una carpeta que posee a su vez otras carpetas llamadas <code>contacts</code> , <code>logs</code> y <code>overrides</code> |
| server.cfg | Configuración relacionada con servidor de proxy y SIP |
| sip.cfg | Todos los parámetros adicionales que se deseen configurar. Para esto es bueno revisar un manual de Usuario Polycom |

Segundo, en estos teléfonos no se realiza ninguna conexión a *telnet* y hasta el momento no se conoce como reiniciar y enviar los archivos de configuración desde la central Elastix como una acción centralizada. Por ende es necesario hacer lo siguiente en el teléfono Polycom:

- Elegir que el tipo de provisionamiento es TFTP.
- Ingresar la IP de la central Elastix.
- Elegir que lea la configuración por TFTP.
- Reiniciar el teléfono Polycom.

Estos pasos son iguales en todos los modelos Polycom que Elastix soporta. Así mismo como en el caso de los teléfonos marca ATCOM se genera un historial en la base de datos interna de Elastix acerca de la configuración del teléfono Polycom.

Si no hay incidentes entonces ahora nuestro teléfono IP ya está configurado desde la central Elastix y cuando se desee cambiar de extensión solo debemos ir al módulo *Endpoint Configurator*, hacemos un par de clics y listo.

Actualización por TFTP

TFTP es el protocolo usado en este caso para proveer provisionamiento. Significa Trivial FTP y es un protocolo que no necesita identificarse, es decir que no necesitamos ni un usuario o contraseña para realizar algún proceso. La mayoría de teléfonos IP proveen este tipo de provisionamiento para poder transferir los archivos de configuración desde la central Elastix hasta los *endpoints* y viceversa, como el caso de los teléfonos Polycom que lo usan para bajar la lista de contactos y escribir logs.

En la central Elastix ya viene configurado el servicio encargado de escuchar para aceptar comunicación o transferencia por TFTP.

Es en el archivo de configuración `/etc/xinetd.d/tftp` donde se personaliza el directorio para uso de tftp. En nuestro caso es `/tftpboot/`.

Personalizando los archivos de configuración

Existen ocasiones en que es necesario personalizar ciertos parámetros en los archivos de configuración para los teléfonos IP. Se piensa hacer una interfaz para ello en versiones futuras, pero para alguien que tiene conocimientos de Linux podrá hacerlo editando los archivos plantillas de configuración que están en la ruta de la central Elastix `/var/www/html/modules/endpoint_configuration/libs/vendors/`

En esta ruta vamos a encontrar lo siguiente.

| Archivo | Descripción |
|-----------------|--|
| atcom.cfg.php | Se encuentra configuración para los modelos AT-530 y AT-320. Para cambiar alguna configuración se recomienda revisar manuales del archivos de configuración para ATCOM |
| polycom.cfg.php | Aquí se encuentran las configuraciones para algunos modelos Polycom. Ya que el contenido de los archivos son XML esto hace que se puedan usar el mismo XML para algunos modelos. |
| linksys.cfg.php | Se encuentra configuraciones para algunos modelos Linksys. Hay que notar que también tienen formato XML |

| | |
|----------------------------------|--|
| cisco.cfg.php, aastra.cfg.php | Estos archivos son versiones aun no terminadas, para la configuración de Cisco y Aastra en un futuro cercano |
|----------------------------------|--|

Al hacer estos cambios en estos archivos plantilla, la próxima vez que use el módulo *Endpoint Configuration* su personalización de configuración será reflejada al configurar o actualizar los teléfonos.

10

Elastix: Configuración avanzada

Si espero perderé la audacia de la juventud...

-- Alejandro Magno

10.1 Ruteo de llamadas

Rutas Entrantes

Esta opción permite configurar el destino de las llamadas que ingresan por una determinada troncal. Cuando una llamada ingresa al servidor, Asterisk puede darse cuenta si coinciden el DID ó el CallerID de la troncal ó el número de la troncal en caso de que sea una de tipo ZAP con la información de la llamada entrante.

Add Incoming Route

Add Incoming Route

Description:

DID Number:

Caller ID Number:

Fax Handling

Fax Extension:

Fax Email:

Fax Detection Type:

Pause After Answer:

Privacy

Privacy Manager:

Options

Alert Info:

CID name prefix:

Music On Hold?

Signal RINGING: ☐

CID Lookup Source

Source:

Set Destination

☐ Announcements:

☐ IVR:

☐ Terminate Call:

☐ Extensions:

☐ Voicemail:

☐ Queues:

Formulario de creación de nueva ruta entrante

DID Number:

Para un *peer* de tipo SIP ó IAX, el DID es proporcionado por el proveedor y generalmente esta asociado al número de cuenta. Por ejemplo si su cuenta es '85679432', poniendo este dato en el campo DID, las llamadas entrantes deberían *"matchear"* esta información y dirigirse al destino configurado.

Caller ID Number:

El CID Number es el Caller ID que es recibido por Elastix. Este dato no es confiable ya que es fácilmente *"spoofable"*. Al dejarlo en blanco *"matchea"* cualquier Caller ID.

Zaptel Channel:

En esta opción se puede colocar el número de la troncal zap que se quiere que se *matchee* con el número de troncal por donde asterisk recibe una llamada. Esta opción es excluyente con el la opción de DID Number.

Fax Handling:

Con esta opción se pueden administrar la forma en que los faxes son recibidos.

Immediate Answer:

Esta característica hace que se conteste la llamada tan pronto como este renguea (con las líneas zaptel, esto ocurre después de que el Caller-ID es recibido, lo cual puede ser después de 3 rings). Si estamos usando g729 ó GSM, los rings debería sonar divertido al llamador.

Pause Alter Answer:

En esta opción se define el número de segundos que deberíamos esperar después de una inmediata respuesta. El primer propósito para esta opción fue la de pausar y escuchar por un tono de fax antes de permitir que la llamada continúe.

Alert Info:

ALERT_INFO puede ser usado para rings distintivos con algunos de dispositivos SIP.

Set destination:

Mediante el *"set destination"* podemos especificar donde queremos que la llamada sea terminada, siendo esto: un IVR, una extensión, un miscellaneous destination, colgar la llamada ó a un contexto personalizado.

Rutas Salientes

Mediante las rutas salientes podemos indicar por que troncal ó troncales deben ser enviadas las llamadas.

Add Route

Route Name:

Route Password:

PIN Set:

None ▾

Emergency Dialing:

☐

Intra Company Route:

☐

Music On Hold?

default ▾

Dial Patterns

Clean & Remove duplicates

Dial patterns wizards:

(pick one) ▾

Trunk Sequence

Submit Changes

Formulario de creación de nueva ruta saliente

Route Name:

En este campo se especifica un nombre descriptivo para la troncal para poder facilitar la administración de las mismas.

Route Password:

Si llenamos este campo cada vez que una llamada salga por una troncal, al llamador será solicitado un password. Si el password es ingresado correctamente la llamada es conectada, caso contrario la llamada es descartada.

Emergency Dialling:

Con esta opción podemos especificar que la ruta es para llamadas de emergencia. Se puede especificar un diferente caller ID.

Dial Patterns:

El patrón de marcado es el conjunto de dígitos ó patrón de dígitos que Asterisk usa para verificar el “match” con los dígitos marcados por un llamador para determinar el canal por donde debe enviar la llamada.

Existen reglas de cómo especificar los patrones de marcados, las cuales indicamos a continuación:

| Patrón | Descripción |
|---------------|---|
| X | Representa cualquier dígito de 0-9 |
| Z | Representa cualquier dígito de 1-9 |
| N | Representa cualquier dígito de 2-9 |
| [1237-9] | Representa cualquier dígito entre corchetes |
| . | Representa uno o más caracteres |
| | Separa el número ubicado a la izquierda del número marcado. Por ejemplo: 9 NXXXXXX debería representar los números marcados como “92234567” pero sólo debería pasar “2234567” |

Trunk Sequence:

Se lista e indica el orden en que se debe intentar el uso de las troncales. Cuando un número digitado por un llamador es “matcheado” por el patrón de marcado especificado antes, Asterisk intentará realizar la llamada por las troncales especificadas en esta opción en el orden en que son listadas.

10.2 Troncales

Una troncal es aquella que permite llevar una llamada a cualquier proveedor de servicio de voz ó a cualquier dispositivo que reciba su intento de llamada y la gestione a otro destino. Alguno de los tipos de troncales son:

- ZAP
- IAX2

- SIP
- Custom

Veamos algunos de los parámetros a configurar en una troncal ZAP.

Add ZAP Trunk

General Settings

Outbound Caller ID:

Never Override CallerID: ☐

Maximum Channels:

Disable Trunk: ☐ Disable

Monitor Trunk Failures: ☐ Enable

Outgoing Dial Rules

Dial Rules:

Dial Rules Wizards:

Outbound Dial Prefix:

Outgoing Settings

Zap Identifier (trunk name):

Creación de nueva troncal tipo ZAP

Outbound Caller ID:

En esta opción especificamos el caller ID con el que se quiere que salga una llamada que es colocada en esta troncal.

El formato utilizado es:

“Nombre del llamador” <#####>

Si no se quiere usar un caller ID dejar en blanco esta opción

Maximum channels:

Esta opción limita el número de canales que pueden ser usados en una troncal. Al decir canales se refiere al número de llamadas simultáneas que se pueden hacer, tanto entrantes como salientes.

Dial Rules:

Las reglas de marcado indican como asterisk debería marcar para hacer la llamada en una troncal. Esta opción puede ser usada para adicionar ó remover prefijos. Si un número marcado no *matchea* con patrones colocados en estas reglas, es marcado como fue digitado.

Las reglas pueden ser especificadas mediante la siguiente sintaxis:

| Patrón | Descripción |
|----------|---|
| X | Representa cualquier dígito de 0-9 |
| Z | Representa cualquier dígito de 1-9 |
| N | Representa cualquier dígito de 2-9 |
| [1237-9] | Representa cualquier dígito entre corchetes |
| . | Representa uno o más caracteres |
| | Separa el número ubicado a la izquierda del número marcado. Por ejemplo: 9 NXXXXXX debería representar los números marcados como “92234567” pero sólo debería pasar “2234567” |
| + | Adiciona un prefijo al número marcado. Por ejemplo: 001+NXXNXXXXXX debería agregar el 001 al número 3058293438 cuando este es marcado |

Troncales de tipo IAX y SIP

Además de las configuraciones arriba especificadas, se deben especificar configuraciones entrantes y salientes que generalmente son especificadas por el proveedor de servicios de voz. Las configuraciones pueden incluir username, password, *codec* utilizado, el *host* con el cual debe de registrarse, etc.

Ejemplo de configuración de los detalles del Peer:

```
allow=ulaw&alaw&gsm&ilbc&g726&adpcm&lpc10
canreinvite=no
disallow=all
dtmfmode=info
fromuser=myusername
host=nyc.voicepulse.com
qualify=5000
secret=mypassword
sendrpid=yes
type=peer
username=myusername
```

Además es posible que se requiera una cadena de registro que tiene la siguiente sintaxis:

```
myusername:mypassword@host
```

Troncales de tipo ZAP

Las troncales de tipo zap son las asociadas a hardware de telefonía instalado en la máquina y usa el módulo `chan_zap.so`. Los archivos asociados a este tipo de troncales son el `/etc/zaptel.conf` y `/etc/asterisk/zapata.conf`

Las troncales Zap son creadas mediante un número asociado a la posición del canal en el hardware.

Troncales personalizadas.

Estas troncales son creadas cuando se usan protocolos no estándares como h323, Unicall, etc.

Se debe especificar un strip de marcado. Por ejemplo:

```
Unicall/XXXXXXX
```

10.3 Tarificación

La opción “Tarifas” del Menú “Facturación” del Elastix nos permite visualizar y configurar las tarifas que se usarán para la facturación de llamadas.



| Prefix | Name | Rate | Rate Offset | Trunk | View |
|--------|-----------------|-------|-------------|----------------|----------------------|
| 001 | Inter2 | 0.020 | 0.100 | SIP/VP-SIP-NYC | View |
| 00 | Internacional | 0.020 | 0.000 | SIP/VP-SIP-NYC | View |
| 08 | Porta de prueba | 0.050 | 0.000 | ZAP/g1 | View |
| 001800 | test | 0.020 | 0.100 | SIP/VP-SIP-NYC | View |

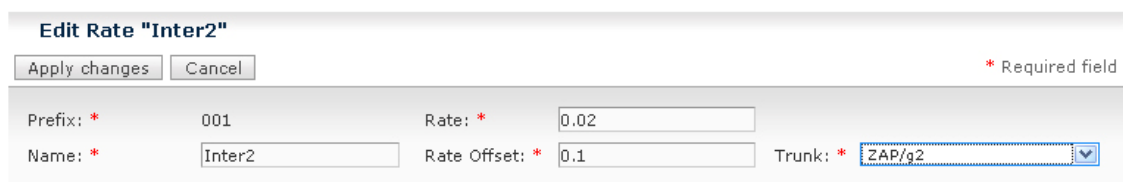
Listado de tarifas definidas

Visualizar, Editar y Eliminar

El enlace Ver nos llevará a una pantalla donde se mostrará los datos de la tarifa.

Para eliminar una tarifa de clic en el botón Eliminar.

Puede además modificar los valores de la tarifa dando clic en el botón Editar. Los únicos valores que puede cambiar son: Nombre, Tarifa y Tarifa de Conexión.



Edit Rate "Inter2"

Apply changes Cancel * Required field

Prefix: * 001 Rate: * 0.02

Name: * Inter2 Rate Offset: * 0.1 Trunk: * ZAP/g2

Formulario de edición de tarifa

Crear Tarifa

Para agregar una nueva tarifa dar clic en el botón Crear Nueva Tarifa. Se mostrará un formulario en el que se ingresarán los siguientes campos:

- **Prefijo:** Prefijo telefónico al cual se aplicará la tarifa
- **Nombre:** Nombre para la tarifa
- **Tarifa:** Valor de la tarifa
- **Tarifa de Conexión:** Valor de conexión

New Rate

Save

Cancel

* Required field

Prefix: *

Rate: *

Name: *

Rate Offset: *

Trunk: *

None

Creación de nueva tarifa

Reporte de Facturación

La opción “Reporte de Facturación” del Menú “Facturación” del Elastix nos permite visualizar el detalle de las llamadas con su respectivo costo y tarifa aplicada.

El reporte cambiará dependiendo de los valores de filtrado:

- **Fecha Inicio:** Fecha a partir de la cual se seleccionarán las llamadas.
- **Fecha Fin:** Fecha hasta la cual se seleccionarán las llamadas.
- **Campo:** Campos adicionales de filtrado: Fuente, Destino, Canal Destino.

Billing Report

Start Date: *

20 Oct 2008

End Date: *

20 Oct 2008

Field:

Destination

Filter

Export

Start

Previous (1 - 50 of 285)

Next

End

| Date | Source | Destination | Dst. Channel | Duration in seconds | Cost | Summary Cost | Rate Applied |
|---------------------|--------|--------------|--------------|---------------------|-------|--------------|--------------|
| 2008-10-20 08:25:52 | 405 | 2215804 | Zap/6-1 | 29 | 1.242 | 1.242 | default |
| 2008-10-20 08:39:33 | 213 | 5001160 | Zap/6-1 | 89 | 1.742 | 2.984 | default |
| 2008-10-20 08:42:47 | 222 | 2347547 | Zap/6-1 | 75 | 1.625 | 4.609 | default |
| 2008-10-20 09:23:47 | 201 | 3810000 | Zap/6-1 | 77 | 1.642 | 6.251 | default |
| 2008-10-20 09:25:44 | 201 | 099456003 | Zap/12-1 | 0 | 1.000 | 7.251 | default |
| 2008-10-20 09:25:53 | 201 | 3810000 | Zap/6-1 | 193 | 2.608 | 9.859 | default |
| 2008-10-20 09:40:24 | 225 | 2445161 | Zap/10-1 | 44 | 1.367 | 11.226 | default |
| 2008-10-20 09:42:12 | 506 | 6013360 | Zap/6-1 | 5 | 1.042 | 12.268 | default |
| 2008-10-20 09:42:29 | 506 | 6013360 | Zap/6-1 | 4 | 1.033 | 13.301 | default |
| 2008-10-20 09:42:51 | 506 | 6013360 | Zap/6-1 | 4 | 1.033 | 14.334 | default |
| 2008-10-20 09:43:21 | 506 | 6013360 | Zap/6-1 | 4 | 1.033 | 15.367 | default |
| 2008-10-20 09:44:23 | 213 | 022529892 | Zap/12-1 | 13 | 1.108 | 16.475 | default |
| 2008-10-20 09:44:18 | 506 | 2631330 | Zap/6-1 | 68 | 1.567 | 18.042 | default |
| 2008-10-20 09:45:34 | 506 | 2631330 | Zap/6-1 | 67 | 1.558 | 19.6 | default |
| 2008-10-20 09:47:25 | 215 | 022447104 | Zap/12-1 | 6 | 1.050 | 20.65 | default |
| 2008-10-20 09:46:49 | 506 | 2631330 | Zap/6-1 | 51 | 1.425 | 22.075 | default |
| 2008-10-20 09:48:01 | 215 | 334022447104 | Zap/12-1 | 11 | 1.092 | 23.167 | default |
| 2008-10-20 09:48:09 | 506 | 6013360 | Zap/6-1 | 3 | 1.025 | 24.192 | default |
| 2008-10-20 09:48:19 | 506 | 6013360 | Zap/6-1 | 3 | 1.025 | 25.217 | default |
| 2008-10-20 09:50:10 | 506 | 097714313 | Zap/12-1 | 15 | 1.125 | 26.342 | default |
| 2008-10-20 09:51:22 | 506 | 6013360 | Zap/6-1 | 2 | 1.017 | 27.359 | default |
| 2008-10-20 09:51:34 | 506 | 6013360 | Zap/6-1 | 4 | 1.033 | 28.392 | default |
| 2008-10-20 09:51:50 | 506 | 6013360 | Zap/6-1 | 3 | 1.025 | 29.417 | default |
| 2008-10-20 09:52:01 | 506 | 6013360 | Zap/6-1 | 4 | 1.033 | 30.45 | default |
| 2008-10-20 09:52:19 | 506 | 6013360 | Zap/6-1 | 2 | 1.017 | 31.467 | default |
| 2008-10-20 09:52:31 | 506 | 6013360 | Zap/6-1 | 2 | 1.017 | 32.484 | default |
| 2008-10-20 09:52:42 | 506 | 6013360 | Zap/6-1 | 4 | 1.033 | 33.517 | default |
| 2008-10-20 09:52:55 | 506 | 6013360 | Zap/6-1 | 4 | 1.033 | 34.55 | default |
| 2008-10-20 09:53:09 | 506 | 6013360 | Zap/6-1 | 2 | 1.017 | 35.567 | default |

Reporte de tarificación

Distribución de Destinos

La opción “Distribución de Destinos” del Menú “Facturación” del Elastix nos permite visualizar mediante un gráfico la distribución de las llamadas salientes agrupadas por tarifa. El gráfico cambiará dependiendo de los valores de filtrado:

- **Fecha Inicio:** Fecha a partir de la cual se seleccionarán las llamadas
- **Fecha Fin:** Fecha hasta la cual se seleccionarán las llamadas
- **Criterio de distribución:** Distribución por Tiempo, Distribución por Número de Llamadas, Distribución por Costo

Distribución por Tiempo

La opción de distribución por tiempo muestra un gráfico de pastel del tiempo traficado para cada troncal. Además muestra una lista de las troncales que más tiempo han traficado.

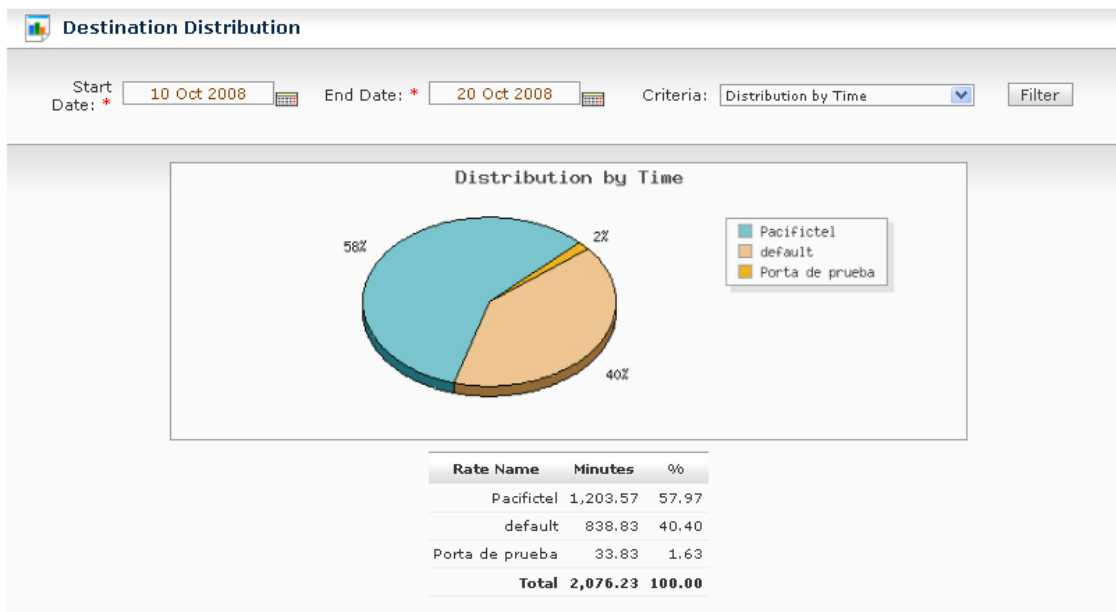
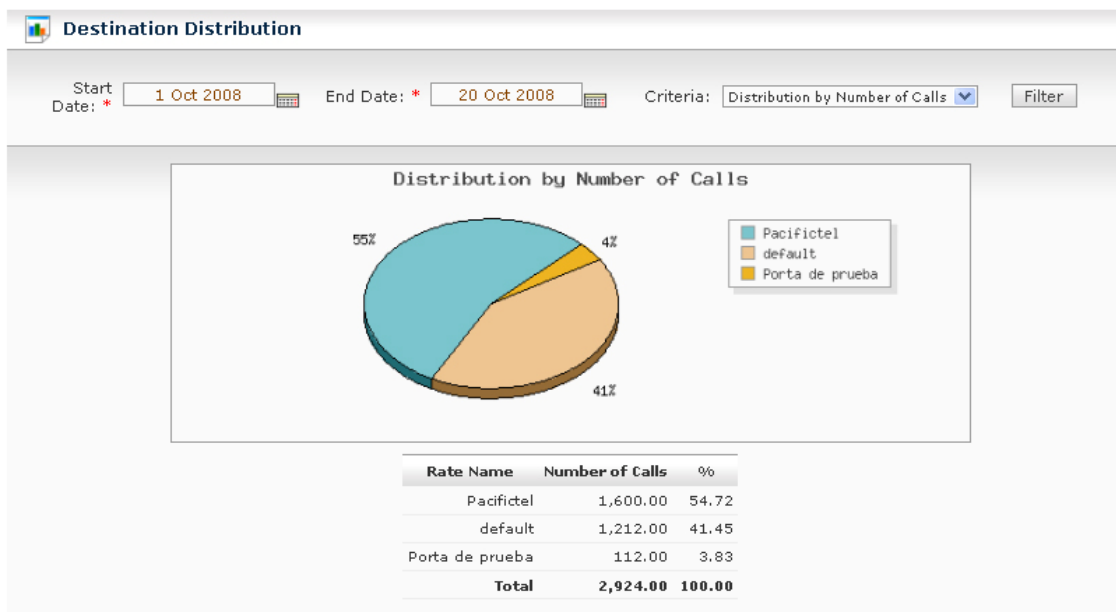


Gráfico de pastel que muestra la distribución de destinos en base a tiempo de uso

Distribución por Número de Llamadas

Muestra un gráfico de pastel del número de llamadas para cada troncal. Además muestra una lista de las troncales con más número de llamadas traficadas.



Distribución de destinos en base a número de llamadas

Distribución por Costo

Muestra un gráfico de pastel dividido por el costo resumido por troncal. Es decir que para cada troncal se computa el costo de todas las llamadas traficadas. Este cálculo de costo está basado en las tarifas ingresadas en el menú “Tarifas” (*rates*).

Configurar Facturación

Dentro de esta opción tenemos dos secciones que explicaremos a continuación.

Configuración de Tarifa por Omisión

La opción “Configuración de Tarifa por Omisión” del Menú “Facturación” del Elastix nos permite determinar el costo por minuto y de conexión para la ruta por omisión o dicho de otra manera la tarifa que se aplicará cuando no se encuentre otra tarifa que aplicar a una llamada.

The screenshot shows two configuration sections. The first section, 'Default Rate Configuration', has an 'Edit' button and a table with two rows: 'Default Rate: *' with a value of '0.50' and 'Default Rate Offset: *' with a value of '1'. A red asterisk and the text '* Required field' are visible in the top right of this section. The second section, 'Trunk Bill Configuration', has a 'Billing Capable' button.

| Default Rate Configuration | |
|----------------------------|------|
| Default Rate: * | 0.50 |
| Default Rate Offset: * | 1 |

| Trunk Bill Configuration | |
|--------------------------|--|
| Billing Capable | |

Tarifa por omisión

Configuración de Troncales para Facturación

La opción “Configuración de Troncales” del Menú “Facturación” del Elastix nos permite determinar cuáles serán las troncales que se usarán para obtener las llamadas para el proceso de facturación.

El listado muestra todas las troncales registradas, deberá seleccionar las que se usarán para la facturación y luego dar clic en el botón “Habilitar para Facturar”.

| | |
|------------------------|------|
| Default Rate: * | 0.50 |
| Default Rate Offset: * | 1 |

| Trunk Bill Configuration | |
|---|--------------------|
| Billing Capable | |
| // Start / Previous (1 - 12 of 12) Next ▶ End ≡ | |
| Trunk | |
| <input checked="" type="checkbox"/> | ZAP/g0 |
| <input checked="" type="checkbox"/> | ZAP/g1 |
| <input checked="" type="checkbox"/> | ZAP/g2 |
| <input checked="" type="checkbox"/> | IAX2/NuFoneIAX |
| <input checked="" type="checkbox"/> | IAX2/rhino |
| <input checked="" type="checkbox"/> | SIP/LargaDistancia |
| <input checked="" type="checkbox"/> | SIP/VP-SIP-NYC |
| <input checked="" type="checkbox"/> | SIP/VP-SIP-SFO |
| <input checked="" type="checkbox"/> | SIP/guayaquil |
| <input checked="" type="checkbox"/> | IAX2/apalosantouio |
| <input type="checkbox"/> | IAX2/out160 |
| <input checked="" type="checkbox"/> | IAX2/out-peru |
| // Start / Previous (1 - 12 of 12) Next ▶ End ≡ | |

Centrales habilitadas para tarificación seleccionadas con visto

10.4 Reportes

Reporte CDR

CDR es la abreviación de *Call Detail Record* o Registro de Detalle de Llamada.

La opción “Reporte CDR” del Menú “Reportes” del Elastix nos permite visualizar un listado con el detalle de las llamadas.

CDR Report List

Start Date: *10 Oct 2008

Field: Destination2

Filter

Delete

End Date: *20 Oct 2008

Status: ALL

Export

StartPrevious (151 - 200 of 4409)NextEnd

| Date | Source | Destination | Src. Channel | Código de Cuenta | Dst. Channel | Status | Duration |
|---------------------|--------|-------------|--------------------------------|------------------|--------------------------------|-----------|----------|
| 2008-10-13 10:16:43 | 212 | 201 | Local/201@from-internal-eb35,2 | | SIP/201-08dba9d0 | NO ANSWER | 0 |
| 2008-10-13 10:17:32 | | 2000 | Zap/10-1 | | Local/202@from-internal-e97b,1 | ANSWERED | 47 |
| 2008-10-13 10:17:49 | 225 | 2305000 | SIP/225-b7d34b18 | | Zap/7-1 | ANSWERED | 249 |
| 2008-10-13 10:30:33 | 225 | 2305000 | SIP/225-b7d41c28 | | Zap/7-1 | ANSWERED | 148 |
| 2008-10-13 10:33:45 | 215 | 201 | SIP/215-b7d41c28 | | SIP/201-08da9578 | ANSWERED | 28 |
| 2008-10-13 10:17:02 | 216 | 2517100 | SIP/216-b7d33588 | | Zap/6-1 | ANSWERED | 1193 |
| 2008-10-13 10:39:37 | 225 | 2683150 | SIP/225-b7d2f610 | | Zap/6-1 | ANSWERED | 60 |
| 2008-10-13 10:50:59 | 506 | 2313103 | SIP/506-b7d22628 | | Zap/6-1 | ANSWERED | 39 |
| 2008-10-13 10:49:05 | 704 | 201 | IAx2/palosantouio-7212 | | SIP/201-b7b15e88 | ANSWERED | 183 |
| 2008-10-13 10:51:34 | 215 | 2275423 | SIP/215-b7d336c0 | | Zap/7-1 | ANSWERED | 74 |
| 2008-10-13 10:51:45 | 506 | 2313103 | SIP/506-b7d34c50 | | Zap/6-1 | ANSWERED | 87 |
| 2008-10-13 10:53:58 | | 502 | Local/502@from-internal-96d5,2 | | SIP/502-08da16d8 | NO ANSWER | 0 |
| 2008-10-13 10:55:07 | 216 | 2517100 | SIP/216-b7d336c0 | | Zap/7-1 | NO ANSWER | 0 |
| 2008-10-13 10:55:09 | 216 | 201 | SIP/216-b7d291f8 | | SIP/201-08da16d8 | ANSWERED | 9 |
| 2008-10-13 11:00:23 | 201 | 2310305 | SIP/201-b7d34b18 | | Zap/7-1 | ANSWERED | 16 |
| 2008-10-13 11:00:47 | 201 | 216916 | SIP/201-b7d06420 | | Zap/7-1 | NO ANSWER | 0 |
| 2008-10-13 11:00:56 | 201 | 2169163 | SIP/201-b7d079b0 | | Zap/7-1 | ANSWERED | 96 |
| 2008-10-13 10:58:21 | 506 | 2646878 | SIP/506-b7d30ba0 | | Zap/6-1 | ANSWERED | 294 |
| 2008-10-13 11:03:43 | 506 | 2295704 | SIP/506-b7d2f610 | | Zap/6-1 | ANSWERED | 89 |
| 2008-10-13 11:12:59 | | 203 | Local/203@from-internal-334a,2 | | SIP/203-08d37ce0 | NO ANSWER | 0 |
| 2008-10-13 11:12:59 | | 202 | Local/202@from-internal-e6f9,2 | | SIP/202-08da16d8 | NO ANSWER | 0 |
| 2008-10-13 11:12:59 | | 502 | Zap/10-1 | | SIP/502-08da16d8 | ANSWERED | 39 |
| 2008-10-13 11:13:28 | 225 | 2512892 | SIP/225-b7d22628 | | Zap/7-1 | ANSWERED | 22 |
| 2008-10-13 11:14:58 | | 502 | Zap/10-1 | | SIP/502-08dba9d0 | ANSWERED | 40 |

Reporte de CDRs

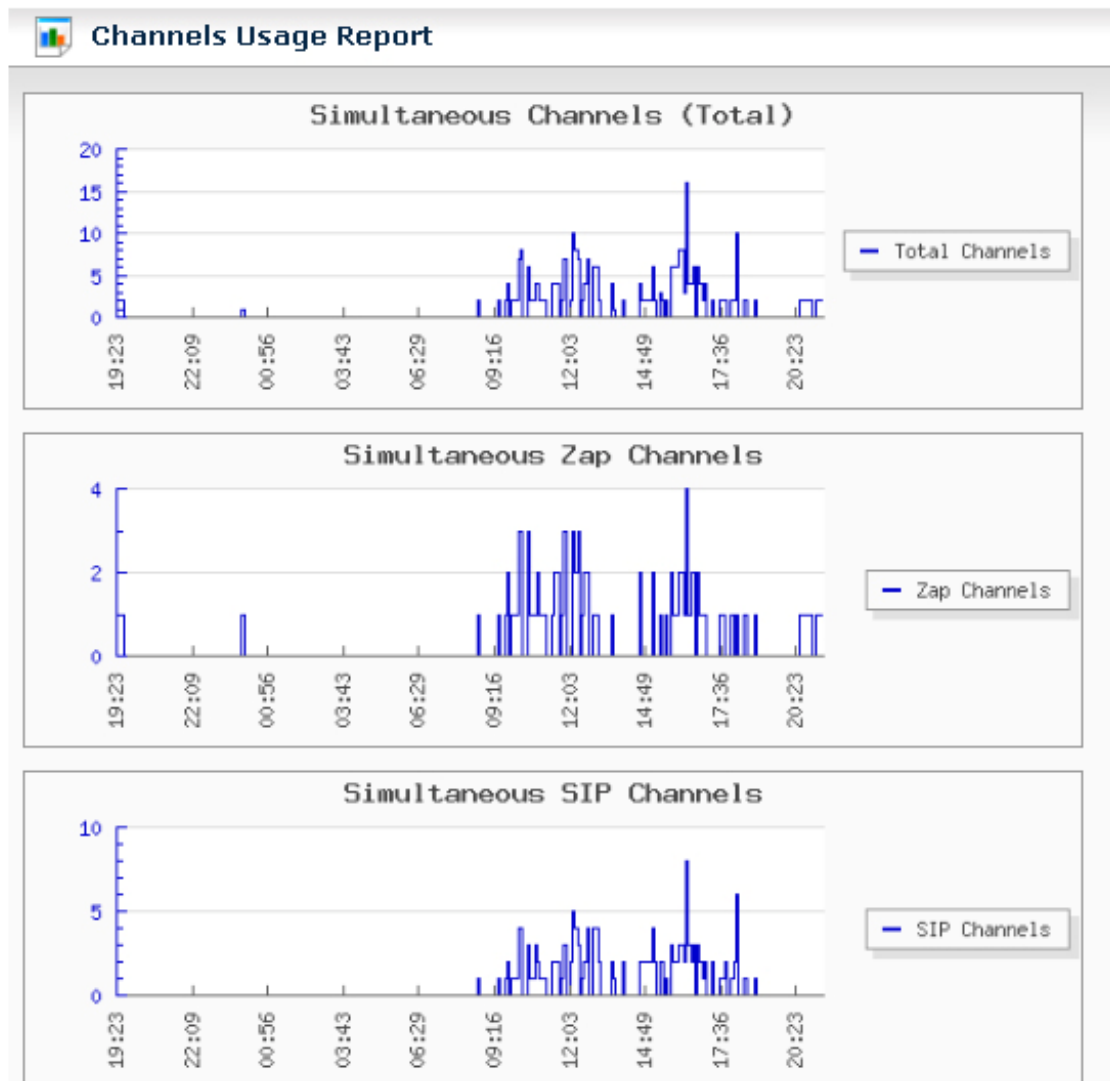
El reporte cambiará dependiendo de los valores de filtrado:

- **Fecha Inicio:** Fecha a partir de la cual se seleccionarán las llamadas
- **Fecha Fin:** Fecha hasta la cual se seleccionarán las llamadas
- **Campos adicionales de filtrado:** Fuente, Destino, Canal Destino
- **Estado** **Estado de la llamada:** ANSWERED, BUSY, FAILED, NO ANSWER

***Nota:** Este reporte es exportable a formato CSV (separado por comas). Este formato se puede abrir desde una hoja de cálculo para realizar las manipulaciones matemáticas que se estime conveniente como por ejemplo sacar promedios de tiempo de llamada por extensión.*

Uso de Canales

La opción “Uso de Canales” del Menú “Reportes” del Elastix nos permite visualizar gráficamente el número de llamadas simultáneas para cada canal.




Reporte de uso de canales

10.5 Configuración DISA

El concepto de DISA (*Direct Inward System Access*) permite que los usuarios puedan acceder telefónicamente al servidor Elastix desde fuera y obtener un tono "interno" de marcado. Con este tono "interno" puedo tener acceso a las mismas facilidades que desde dentro. Por ejemplo, podría realizar llamadas internacionales (si es que se me lo permite claro está).

Normalmente el soporte DISA no está disponible para todos los empleados de una compañía y es necesaria una clave para poder obtener el tono "interno". Para poder ingresar la clave algunas compañías proveen de una opción oculta en el IVR principal de la compañía. En todo caso hay que tomar las precauciones para que el acceso DISA no pueda ser quebrantado por usuarios indeseados, quienes podrían aprovecharlo para hacer llamadas a costa de la compañía.

Para configurar un acceso DISA debemos ir al menú "PBX -> PBX Configuration -> DISA" y obtendremos una pantalla como la siguiente.



The screenshot shows a web form titled "Add DISA". Below the title is a horizontal line. The form contains several input fields and a checkbox, each with a label to its left: "DISA name:" with the value "disa-test", "PIN" with the value "3434", "Response Timeout" with the value "10", "Digit Timeout" with the value "5", "Require Confirmation" with an unchecked checkbox, "Caller ID" with an empty field, and "Context" with the value "from-internal". At the bottom left of the form is a button labeled "Submit Changes".

Formulario para crear un nuevo acceso DISA

- **DISA name:** El nombre que se le da al nuevo acceso DISA. Puede ser cualquiera
- **PIN:** La clave para que el usuario acceda al soporte DISA (tono interno)
- **Response Timeout:** Tiempo máximo que la aplicación DISA espera para que el usuario complete el PIN
- **Digit Timeout:** Máximo tiempo entre dígitos

Luego, para que éste acceso DISA esté disponible desde el IVR principal debemos modificar dicho IVR y asociar una opción con el acceso DISA recientemente creado.

10.6 Condiciones de tiempo

El uso de condiciones de tiempo es una característica muy útil para muchos negocios que quieren modificar el comportamiento de su PBX dependiendo de la hora del día o día de la semana. Por ejemplo, la compañía X desea que en la noche las llamadas a ventas no vayan a la cola de ventas sino que se reproduzca un mensaje indicando el horario de atención y diciendo también que pueden contactar a un número alternativo para casos urgentes; o también la compañía Z que quiere reproducir un IVR de bienvenida distinto al habitual durante los fines de semana ya que tienen promociones de fin de semana.

Las condiciones de tiempo son funcionalidad provista gracias a la integración de Elastix con el software freePBX. Para configurar una condición de tiempo hay que ir al menú "*PBX -> PBX Configuration -> Time Condition*". Aquí accederemos a un formulario que nos permitirá establecer el rango de tiempo asociado con la condición que queremos crear. Al final del formulario podemos relacionar esta condición de tiempo con una entidad como por ejemplo un IVR, una extensión, una cola, un *voicemail*, etc.

Add Time Condition

Add Time Condition

Time Condition name:

Time to match:

Time to start: - : -

Time to finish: - : -

Week Day Start: -

Week Day finish: -

Month Day start: -

Month Day finish: -

Month start: -

Month finish: -

Destination if time matches:

- ☒ Announcements:
- ☐ IVR:
- ☐ Terminate Call:
- ☐ Extensions:
- ☐ Voicemail:
- ☐ Queues:
- ☐ Misc Destinations:
- ☐ Phonebook Directory:
- ☐ Conferences:

Destination if time does not match:

- ☐ Announcements:
- ☐ IVR:
- ☐ Terminate Call:

Añadiendo una nueva condición de tiempo

11

Funcionamiento interno de freePBX

La libertad no se implora, se conquista

-- Eloy Alfaro

11.1 Fundamentos de freePBX

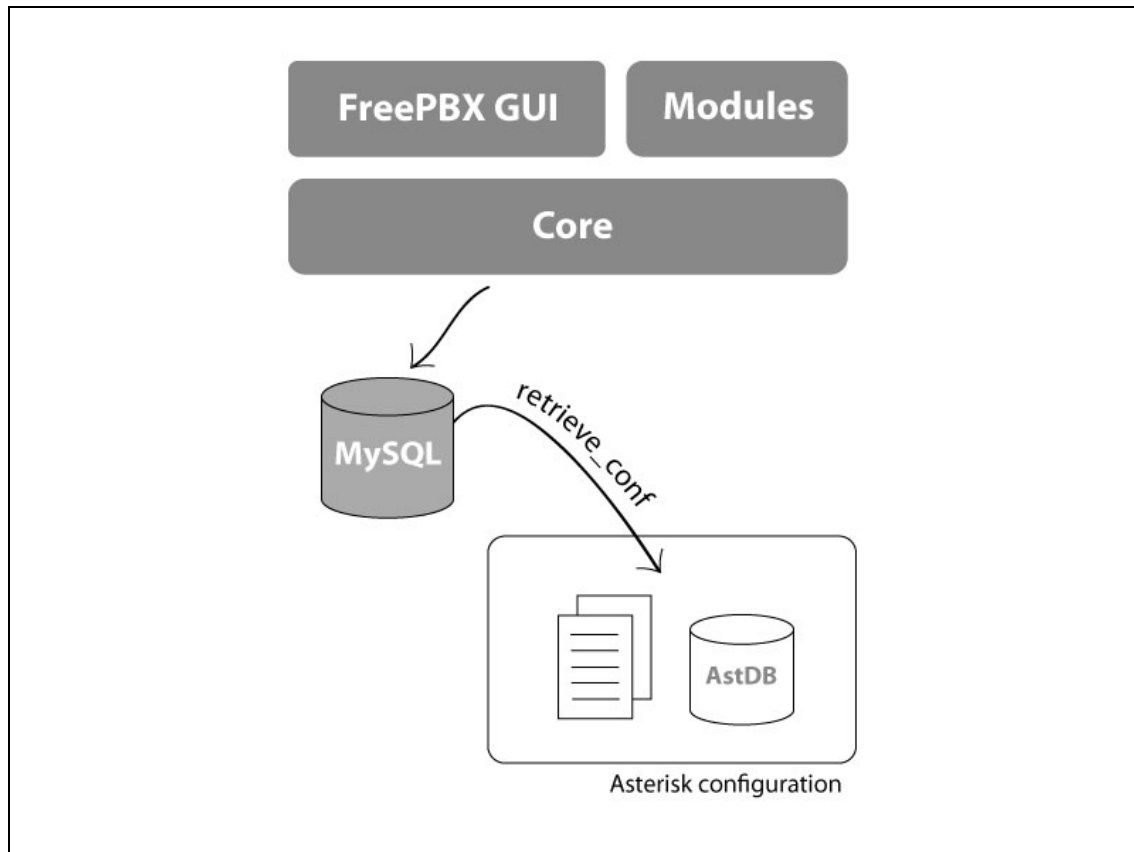
freePBX es un poderoso software de configuración de Asterisk que ha sido integrado como parte de la distro Elastix. Básicamente es el software que se encarga de generar los archivos de configuración de Asterisk incluido el plan de marcado en base a la configuración ingresada por el usuario desde el Web.

Para realizar esta difícil tarea freePBX la divide en dos pasos:

1. Recaba la configuración deseada por el usuario en una base de datos de configuración. En el caso de Elastix se usa el motor de base de datos MySQL.

2. Luego convierte esa información almacenada en MySQL principalmente en archivos de configuración a través de una herramienta de *parsing* llamada `retrieve_conf`

Para ilustrar mejor el proceso los dejamos con la siguiente figura.



Componentes principales de freePBX

11.2 Organización de archivos

freePBX escribe o lee información de algunos directorios para su correcto funcionamiento. A continuación una lista de los más importantes.

| Directorio | Descripción |
|-----------------------------------|--|
| <code>/var/www/html/admin/</code> | El directorio principal donde reside el GUI de freePBX |
| <code>/var/www/html/panel/</code> | Directorio del Flash Operador Panel. En realidad se trata de otro proyecto independiente pero relacionado. |
| <code>/etc/asterisk/</code> | El directorio donde residen los archivos de configuración de Asterisk que escribe |

| | |
|----------------------------------|---|
| | freePBX |
| /var/lib/asterisk/agi-bin/ | Aquí residen los <i>scripts</i> AGI de Asterisk. freePBX cuenta con algunos <i>scripts</i> en esta carpeta como el <code>dialparties.agi</code> |
| /var/lib/asterisk/bin/ | Algunos binarios utilizados por freePBX. Algunos de ellos son links simbólicos a las carpetas de los módulos |
| /usr/local/sbin/ | Aquí reside el <i>script</i> <code>amportal</code> . Se encuentra en esta carpeta porque puede ser manipulado solo por <code>root</code> |
| /var/lib/asterisk/sounds/ | Aquí residen los sonidos que se utilizan en el plan de marcado por freePBX |
| /var/lib/asterisk/sounds/custom/ | Aquí residen los sonidos que han sido creados por el usuario. Por ejemplo cuando se crea un IVR de bienvenida |
| /var/spool/asterisk/voicemail/ | Directorio donde se almacenan los <i>voicemails</i> |
| /var/spool/asterisk/monitor/ | En este directorio se almacenan las grabaciones de voz |
| /var/log/asterisk/cdr-csv/ | Directorio donde se escriben los CDRs en formato texto |

11.3 Configuración

amportal.conf

El archivo `/etc/amportal.conf` es un archivo de texto plano leído por freePBX para recabar información general de configuración. Digamos que es el principal archivo de configuración de freePBX y puede ser editado manualmente en caso de requerirse.

Por ejemplo, aquí se encuentra información de acceso a la base de datos como usuario y clave así como también el tipo de autenticación que freePBX usará para con sus usuarios Web.

Es un archivo bien documentado por lo que no se nos hará difícil entender las opciones o directivas de configuración.

11.4 retrieve_conf: generador del plan de marcado

`retrieve_conf` es un programa escrito en PHP que se encarga de generar el plan de marcado a partir de la información almacenada en la base de datos. Este programa puede ser invocado inclusive desde la línea de comandos aunque usualmente se ejecuta desde el Web a través de un *script* PHP cada vez que damos clic en la famosa barrita roja de freePBX (la que dice “aplicar cambios”).

11.5 dialparties.agi

Se trata de un *script* AGI, escrito en PHP, que se invoca desde el plan de marcado cuando se quiere hacer una llamada. Este *script* se encarga de revisar el estado de la extensión a la que se está tratando de alcanzar, así como también ciertas opciones que pudieran estar configuradas en dicha extensión (como *call waiting*, *call forward*, *do not disturb*) y retorna dicha información para que se pueda tomar una acción acorde como por ejemplo devolver al llamante un tono de ocupado en el caso de DND o rutar la llamada a otra extensión en el caso de *call forward*. Si la extensión está disponible y no se encuentran opciones activadas la llamada transcurre sin inconvenientes.

Para realizar su trabajo `dialparties.agi` consulta la base de datos de Asterisk (la que se encuentra almacenada en el archivo `/var/lib/asterisk/astdb`).

A continuación algunas líneas de diagnóstico que aparecen en el CLI cuando se intenta hacer una llamada. Estas líneas son devueltas por el *script* `dialparties.agi`.

```
dialparties.agi: Caller ID name is 'Pedro Paramo' number is '5454'
dialparties.agi: Methodology of ring is 'none'
-- dialparties.agi: Added extension 1616 to extension map
-- dialparties.agi: Extension 1616 cf is disabled
-- dialparties.agi: Extension 1616 do not disturb is disabled
dialparties.agi: Extension 1616 has ExtensionState: 0
-- dialparties.agi: Checking CW and CFB status for extension 1616
-- dialparties.agi: dbset CALLTRACE/1616 to 5454
-- dialparties.agi: Filtered ARG3: 1616
```

Nota: Este script puede ser sobrescrito por el propio freePBX por uno nuevo por lo que si se realizan cambios en su código se debe tener este hecho en cuenta. `dialparties.agi` está ubicado en la ruta `/var/lib/asterisk/agi-bin/dialparties.agi`.

11.6 Estructura de un módulo de freePBX

La idea de este apartado no es proveer de los conocimientos para que podamos construir un módulo de freePBX sino más bien para que podamos entender cómo solucionar posibles problemas y para entender mejor cómo funciona freePBX.

Los módulos de freePBX tienen la siguiente estructura mínima de archivos.

| Nombre del archivo | Descripción |
|-----------------------------------|---|
| module.xml | Contiene información del módulo en formato XML |
| page. nombredelmodulo .php | Es el archivo principal del módulo escrito en PHP. Como es de imaginar, la palabra nombredelmodulo debe ser reemplazada por el respectivo nombre del módulo. |
| functions.inc.php | Se trata básicamente de una librería de funciones PHP. Funciones tanto para uso del módulo como para que puedan ser compartidas por otros componentes de freePBX |
| install.sql | Contiene sentencias SQL que se ejecutarán sobre la base de datos al instalar el módulo. Básicamente para crear las tablas necesarias para el módulo |
| uninstall.sql | Contiene las sentencias SQL necesarias para desinstalar el módulo. Básicamente para remover las tablas creadas |
| install.php | Script PHP para ser ejecutado al momento de instalación del módulo |
| uninstall.php | Script PHP para ser ejecutado al momento de desinstalación del módulo |

12

Interconexión con Elastix remotos

Cualquier poder, si no se basa en la unión, es débil

-- Jean de la Fontaine

12.1 A través de troncales SIP

Una de las ventajas de usar Elastix es que fácilmente podemos interconectar dos cajas Elastix de tal manera que se puedan unir las comunicaciones de dos lugares remotos, por ejemplo en un solo plan de marcado.

A continuación veremos como configurar la interconexión de dos Elastix mediante una troncal de tipo SIP.

En el escenario mostrado en el gráfico anterior queremos interconectar un servidor Elastix en el punto A con otro servidor Elastix en el punto B de tal forma que el teléfono en A y el teléfono B se puedan comunicar únicamente marcando las extensiones directamente de cada teléfono.

Para implementar esto se requiere definir dos cosas:

1. La troncal
2. Las rutas salientes en cada Elastix

Definiendo la troncal

Definimos el nombre de la troncal el cual en nuestro caso servirá de *username* para el proceso de registro. A la troncal del Elastix en el punto A la llamaremos pointA y la del punto B la llamaremos pointB.

Luego definimos los detalles del *peer* que en este caso serán los siguientes:

```
host=dynamic
secret=welcome
trunk=yes
type=friend
```

Configuración que se repite en ambos servidores Elastix. Como se ve hemos definido como “*secret*” la palabra welcome.

Definimos el *register string* siguiendo el formato especificado anteriormente.

En este caso para el servidor A debería ser:

```
pointB:welcome@192.168.1.240
```

y para la central Elastix en el servidor B debería ser:

```
pointA:welcome@192.168.1.239
```

Una vez establecido esta configuración en ambos Elastix podemos verificar en el Asterisk CLI si se encuentran registradas las troncales mediante el comando: `sip show registry` y deberíamos ver algo como esto:

```
elastix*CLI> sip show registry
Host                Username    Refresh State    Reg.Time
192.168.1.240:5060   pointB      105 Registered   Fri,11 Apr 2008 10:29:43
```

El siguiente paso es definir los detalles del `user`, es decir en donde se quiere terminar la llamada. En este caso queremos que sea accesible a la extensiones local, bastará con colocar el contexto primario. Esto indica que cuando la llamada entre por la troncal iniciará en el contexto aquí especificado. La configuración es como sigue:

```
context=from-internal
host=192.168.1.240
insecure=very
type=friend
```

Definiendo la ruta saliente

- Primero determinamos el nombre de la ruta saliente. En este caso le pondremos: ElastixB
- Luego definimos el patrón de marcado, en este caso usaremos el siguiente patrón de marcado: 2|XXX. Estamos suponiendo que los números locales empiezan con el número 2 adelante. Debemos ajustar esto dependiendo del país donde nos encontremos.
- En tercer lugar escogemos la troncal por la cual debe irse la llamada. En este caso escogemos por la troncal SIP/pointA

12.2 Conexión con proveedores de llamadas vía IP

Hoy en día ha tenido mucho auge lo que se llaman los VSP (*Voice Service Provider*) ó proveedores de servicios de voz, los cuales proveen entre cosas conexiones SIP ó IAX para permitir conectar llamadas locales en una región.

En este apartado implementaremos una conexión SIP con un proveedor que nos permitirá hacer llamadas dentro de los Estados Unidos.

En el escenario mostrado en el gráfico anterior queremos registrar nuestra troncal de tipo SIP con un VSP que nos permita realizar llamadas a USA desde cualquier teléfono IP.

Para implementar esto se requiere definir dos cosas:

1. La troncal

2. Las rutas salientes en cada Elastix

Definiendo la troncal

Como vimos anteriormente se debe definir el nombre de la troncal en este caso le llamaremos: VP-SIP-NYC

Luego definimos las características del *peer*. Esta información debe ser proporcionada por el VSP y generalmente es la misma para todos los usuarios a registrar, lo único que cambia es el username y el password.

En nuestro caso nuestro VSP nos pidió que configuremos de la siguiente forma.

```
allow=ulaw&alaw&gsm&ilbc&g726&adpcm&lpc10
canreinvite=no
disallow=all
dtmfmode=info
fromuser=myusername
host=nyc.voicepulse.com
qualify=5000
secret=mypassword
sendrpid=yes
type=peer
username=myusername
```

Luego agregamos la cadena de registro que en nuestro caso sería la siguiente.

```
myusername:mypassword@nyc.voicepulse.com
```

Definiendo la ruta saliente

- Primero determinamos el nombre de la ruta saliente. En este caso le pondremos: INTERNACIONAL
- Luego definimos el patrón de marcado, en este caso usaremos el siguiente patrón de marcado: 00.
- En tercer lugar escogemos por que troncal debe irse la llamada en este caso escogemos por la troncal SIP/VP-SIP-NYC

Una vez establecido esta configuración podemos verificar en el Asterisk CLI si se encuentra registrada nuestra troncal mediante el comando: `sip show registry` y deberíamos ver algo como esto:

```
elastix*CLI> sip show registry
```

| Host | Username | Refresh State | Reg.Time |
|-------------------------------------|------------|----------------|-----------------|
| nyc.voicepulse.com:5060 10:29:43 | myusername | 105 Registered | Fri,11 Apr 2008 |

Con esto estamos listos para realizar llamadas hacia USA.

13

Calidad de voz

Calidad es hacer las cosas bien cuando nadie está controlando

-- Henry Ford

13.1 Síntomas que afectan la calidad de voz

Enumerar los problemas que afectan la calidad de voz es difícil pues a lo largo de los años me he encontrado con muchos, algunos muy parecidos a otros lo cual hace difícil categorizarlos y cuyas causas son muy variadas. Sin embargo, en el presente apartado pretendo enumerar los más comunes explicando sus causas y posibles soluciones.

Eco

El eco es uno de los síntomas más comunes y es muy fácil reconocerlo. Se produce cuando una parte de la señal de ida se refleja en la señal de vuelta.

Hay algunas causas del eco pero una de las más comunes se produce en las líneas analógicas cuando se combinan las señales en el convertidor híbrido o convertidor de 2 a 4 hilos ya que es muy difícil separar las señales de ida y de vuelta de manera eficaz.

El problema se agrava cuando la impedancia de la línea telefónica varía mucho. Mucha de la tarjetería telefónica disponible para Asterisk no dispone de un buen mecanismo dinámico de ajuste de la impedancia de la línea con la impedancia de la tarjeta. Por esta razón una parte de la onda se refleja.

Nota: hay que recordar que para que un receptor absorba eficientemente la potencia de la señal recibida su impedancia debe coincidir con la de la línea.

Sin embargo, existe una forma de acoplar estas impedancias lo mejor posible. Esto se puede realizar con la utilidad llamada `fxotune` disponible en Elastix. Lamentablemente esta utilidad se debe ejecutar manualmente y con el servicio Asterisk apagado. Para líneas con problemas se recomienda ejecutarlo en horas no laborables una vez por semana. El comando es el siguiente.

```
fxotune -i 5
```

Lo que hace el comando `fxotune` es hacer prueba y error enviando una señal pura por la línea y escuchando el retorno. Esto lo hace muchas veces para cada línea hasta encontrar el mejor valor de ganancia, el cual escribe en un archivo ubicado en `/etc/fxotune.conf`

Al finalizar su ejecución podemos encender Asterisk de nuevo pero antes hay que ejecutar el siguiente comando para decirle a Zaptel que cargue los valores nuevos de ganancia.

```
fxotune -s
```

Esto lo debemos ejecutar siempre antes de arrancar Asterisk por lo que se recomienda mejor incluirlo al final del *script* de arranque de Zaptel ubicado en la ruta `/etc/init.d/zaptel` para no tener que hacerlo manualmente cada vez.

Nota: Es muy importante hacer notar que si se utiliza el `fxotune` para tratar de acoplar las líneas no se deben configurar las variables `txgain` o `rxgain` en el `zapata.conf` ya que sobrescribirán el trabajo del `fxotune`.

Otra causa del eco es el eco acústico provocado cuando la señal de sonido se retroalimenta desde el micrófono al audífono. Esto es más notable cuando hablamos por altavoz. Por supuesto el diseño del teléfono tiene mucho que ver aquí y hay modelos que introducen menos eco acústico que otros.

Bajo nivel o volumen

Muchas redes telefónicas de baja calidad atenúan la señal de manera significativa haciendo que oigamos un volumen muy bajo. Esto obviamente afecta la calidad de voz, haciendo que la conversación muchas veces sea inentendible o haciendo que no se puedan detectar los dígitos DTMFs.

Tanto el volumen de recepción como el de transmisión pueden ser amplificados mediante parámetros en el archivo `zapata.conf`. Sin embargo se debe tener mucho cuidado pues en ocasiones un volumen bajo significa una línea mal acoplada y por ende la mejor solución es utilizar la utilidad `fxotune` para corregir el problema. Si subimos a ciegas los parámetros de ganancia en este caso podríamos conseguir empeorar el eco en la línea.

El parámetro para modificar la ganancia de la señal de transmisión se llama `txgain` y el que sirve para modificar la ganancia de la señal de recepción se llama `rxgain`. Ambos se pueden modificar en el archivo `zapata.conf`.

Retardo

El retardo no es otra cosa que la demora de la voz en llegar a destino. Usualmente el retardo es menor a un segundo y si es menor a 200 ms pasa casi desapercibido.

Retardos mayores a 500 ms provocan que la conversación se pise, es decir que los interlocutores se interrumpan y la conversación se traslape. Esto es irritante para la conversación.

Cuando existe retardo es casi imposible eliminarlo a nivel del servidor pues en la gran mayoría de los casos el retardo es un síntoma de problemas inherentes a la red de comunicaciones. Con esto quiero decir que si se quiere eliminar el retardo habrá que analizar si se puede cambiar o mejorar algo en la red de comunicaciones. Si hablamos de una red de paquetes, puede ser que uno de los equipos (por ejemplo un ruteador) esté saturado en su capacidad.

Distorsión de la voz

En esta categoría recaen diferentes problemas. Sin embargo uno común es el de los usuarios que reportan algo como “se escucha robotizado”.

Cuando se escucha robotizado usualmente se trata de usuarios que utilizan un *codec* ahorrador de ancho de banda como por ejemplo *gsm*. Estos *codecs* realizan un muy buen trabajo comprimiendo la voz lo máximo posible pero el costo es la pérdida de

información en el proceso de codificación. Si a esto se le agregan problemas con el ancho de banda el problema empeora.

La solución en este caso en particular es cambiar de *codec* pero hay que tener presente que eso podría disparar otro problema peor si es que se usa un *codec* mas consumidor de ancho de banda y el enlace se satura.

Comunicación entrecortada

Un problema muy molesto por cierto que normalmente está relacionado con la pérdida de paquetes. A su vez la pérdida de paquetes puede ser causada por diferentes problemas en la red, siendo el más común el de redes con una latencia elevada o ancho de banda limitado. La comunicación entrecortada también puede ser ocasionada por un elevado *jitter* en la red. Estudiaremos esto último más adelante.

En todo caso lo normal es buscar la causa en la red y no en el servidor.

Por lo general si la latencia de la red es siempre (no hablo del promedio) de menos de 150 ms y el canal de comunicaciones no se encuentra saturado podemos estar tranquilos de que los problemas de comunicación entrecortada no nos quitarán el sueño.

13.2 Parámetros relacionados con la calidad de voz en redes de paquetes

Retardo de red

Hay que distinguir aquí que no estamos hablando de retardo de voz sino el retardo de los paquetes de red en las redes de paquetes.

Una manera sencilla (aunque no exacta) de calcular el retardo de la red es utilizar al comando ping. El comando ping nos presenta al final un pequeño resumen estadístico de los paquetes enviados. El resumen luce como el siguiente.

```
--- ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5004ms
rtt min/avg/max/mdev = 73.055/74.181/74.852/0.710 ms
```

Aquí podemos ver algunos parámetros como el retardo mínimo, máximo, promedio y desviación.

Pérdidas de paquetes

Las pérdidas son ocasionadas por paquetes que no llegaron a su destino. Pueden haber muchas razones para esto como equipos defectuosos o saturados, pérdidas en el medio de transmisión (cables mal ponchados, ruido ambiente elevado), etc.

En el reporte del comando ping examinado hace poco vemos que también se nos reporta porcentualmente las pérdidas de paquetes. Lo deseable es que no existan pérdidas de paquetes en lo absoluto. Si existen, hay que averiguar el por qué.

Inclusive pérdidas de menos del 1% pueden afectar a las conversaciones de voz sobre IP. Más aun si usamos *codecs* con gran compresión.

Jitter

El *jitter* es un parámetro muy importante cuando se habla de calidad de voz en redes de paquetes. El *jitter* se define como la variabilidad del retardo y normalmente está en el orden de los milisegundos.

Este parámetro es necesario para analizar la calidad de voz pues conocer que tenemos un promedio de retardo bajo no es suficiente para garantizar una buena calidad. Por qué?

Porque si tenemos un promedio de retardo aceptable pero en cambio es muy variable esto significa que existe más probabilidad que los paquetes lleguen en desorden o con retardos excesivos y debemos recordar que debido a que estamos hablando de comunicaciones en tiempo real no se pueden esperar a que lleguen todos los paquetes, habrá que descartar los que se demoren más de lo necesario. Por tanto, si tenemos un *jitter* elevado es más probable que se descarten paquetes y por lo tanto oigamos una conversación entrecortada.

Nota: Muchas personas confían ciegamente en el comando ping para obtener el retardo de una red e incluso para calcular el jitter haciendo cálculos manuales. Pero hay que notar que, pese a que puede usarse en situaciones en las que no tengamos otra herramienta a la mano, el comando ping no envía paquetes reales de voz sino tráfico ICMP lo cual no es lo mismo y no recibe el mismo tratamiento. Inclusive en ciertas redes este protocolo puede tener una limitación de ancho de banda totalmente diferente al de los paquetes de voz. La manera más exacta de calcular el jitter es con paquetes RTP reales y esto se puede analizar con un sniffer.

Jitter buffers

Para tratar de que el problema del *jitter* tenga un menor impacto en la calidad de voz se pensó en que si se espera un poquito por los paquetes de voz en el destino (antes de re-ensamblar la señal de voz para enviarla al teléfono del destinatario) a lo mejor lleguen una buena parte de los paquetes extraviados y la comunicación sufra menos las consecuencias.

El problema es que con esta acción estamos también aumentando el retardo, que es otro problema desagradable. Sin embargo la idea es buena y si esperamos lo sólo lo justo es probable que lleguen una buena parte de los paquetes perdidos sin afectar significativamente el retardo.

Un *jitter buffer* entonces es un pequeño registro donde se almacenan temporalmente todos los paquetes de voz durante un corto tiempo para esperar por posibles paquetes perdidos. El tiempo de espera se denomina “tamaño del buffer”.

Asterisk 1.4 ya soporta el concepto de *jitter buffer* y este parámetro es configurable a nivel de protocolos IAX y SIP. Este parámetro se puede configurar por tanto en los archivos de tecnología correspondientes (iax.conf y sip.conf).

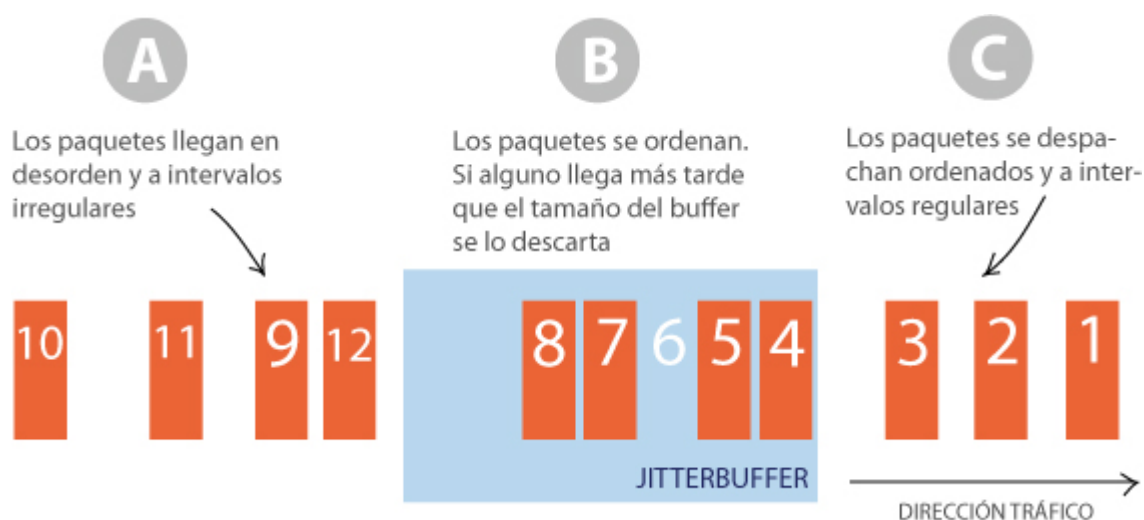
Si se quiere estar seguro de que el *jitter buffer* está habilitado se puede añadir las siguientes opciones en los archivos de tecnología. usando.

En el sip.conf

```
jbenable=yes  
jbmaxsize=200 ; Opcional (define el tamaño del buffer)
```

En el iax.conf

```
jitterbuffer=yes  
maxjitterbuffer=200 ; Opcional (define el tamaño del buffer)
```

Esquema de funcionamiento de un jitterbuffer

13.3 Cómo medir la calidad de voz

Medir la calidad de voz siempre ha sido un rompecabezas para los ingenieros y su complejidad radica en que la calidad de voz es en parte un parámetro subjetivo de la persona que escucha. Incluso muchos conceptos como la entendibilidad de una conversación telefónica varían dependiendo del idioma que usemos. Para ponerlo más complicado aún, nuestra mente juega una serie de trucos con nosotros para “facilitarnos la vida” que hacen que ciertas mediciones de la calidad de voz sean más difíciles de obtener. Un ejemplo de esto es que nuestra mente le da más importancia a eventos más recientes que a los pasados por lo que si la voz se distorsiona justo antes de terminar la conversación es probable que las personas califiquen la calidad de la llamada de manera diferente a que si los errores ocurrieron al principio de la llamada. También hay personas que justifican inconscientemente cierta falta de calidad de voz en pro de las ventajas o conveniencias personales percibidas y un ejemplo de esto es que las personas no se quejan mucho de la calidad de voz en teléfonos celulares (al menos de primera generación) en pro de la ventaja de la movilidad. Sin embargo, si la misma calidad la escucharan en una línea fija es probable que llamen a la compañía telefónica a quejarse porque su línea suena raro o con mala calidad. Dicho de otra manera inconscientemente estamos justificando esa falta de calidad y haciéndola “justificable”.

En fin, esto es solo la punta del iceberg pues hay una serie de parámetros subjetivos un tanto difíciles de cuantificar. Lo importante es ilustrar la difícil tarea que se tiene al tratar de cuantificar la calidad de voz.

Ahora vayamos al grano. En efecto existen algunos intentos de estandarizar la medición de la calidad de voz. Uno de los más conocidos es la famosa escala MOS que se basa en mediciones subjetivas y también se encuentra el modelo E que ya pone en el tapete algunos parámetros objetivos como el retardo o la pérdida de paquetes. Algo interesante de este último modelo es que contempla la conversión de sus resultados a la famosa escala MOS, de manera que podríamos decir que ya tendríamos una escala estándar que podemos usar para calificar la calidad de voz.

Escala MOS

La escala MOS es realmente una recomendación de la ITU. Específicamente la recomendación ITU-T P.800 y describe una escala de calidad de voz basada en la toma de muestras subjetivas que se realizan con una serie de técnicas llamadas ACR (*Absolute Category Rating*).

Para hablar claro se reúne a un grupo grande de personas y se les pide que califiquen la calidad de voz. Las pruebas ACR, a diferencia de otras, no contemplan la comparación de la conversación telefónica con una señal referencial y pensándolo bien así es como sucede en una llamada telefónica real.

Pero al no tener un parámetro de comparación o referencia, cómo podemos normalizar las muestras obtenidas?. Para esto se le hacen escuchar a los participantes algunos ejemplos que previamente han sido definidos por la Recomendación de manera que los usuarios tengan algún marco referencial.

Una vez hecho esto se transmiten una serie de frases (también pre-definidas por la Recomendación) por la línea telefónica y los usuarios proceden a calificar la calidad de voz.

A continuación la escala MOS.

| Calificación MOS | Calidad | Esfuerzo |
|------------------|-----------|---|
| 5 | Excelente | No hace falta esfuerzo alguno |
| 4 | Buena | Es necesario prestar atención pero no es necesario un esfuerzo apreciable |
| 3 | Aceptable | Esfuerzo moderado |
| 2 | Pobre | Gran esfuerzo |
| 1 | Mala | No es posible entender la conversación |

Modelo E

El modelo E es un modelo más matemático y objetivo de medición de la calidad de voz basado en algunos parámetros de red como el retardo, el *jitter* y la pérdida de paquetes.

Nota: Al decir que es un método “objetivo” el lector no debe pensar necesariamente que este es un mejor método de medición de la calidad de voz porque ya hemos expuesto que la calidad de voz contiene un importante componente subjetivo. Sin embargo, si teneos la posibilidad de tomar mediciones de los parámetros de red, este método es más práctico que el

primero pues luego de un relativamente sencillo cálculo matemático podemos tener una aproximación de la calidad de voz en la escala MOS.

El modelo E también fue recomendado por el ITU y se trata de la recomendación ITU-T G.107. Para ir a lo que nos interesa, el modelo E dice que la calidad de voz queda representada por un parámetro R. Luego veremos como traducir ese parámetro R a la escala MOS, pero primero veamos cómo se calcula R.

$$R = R_0 - I_S - I_d - I_e + A$$

Donde,

R_0 es la relación señal-a-ruido (signal-to-noise ratio)

I_S es la degradación por conversión (la degradación que sufre la señal en su conversión a un formato paquetizado para poder transmitirla en una red de paquetes)

I_d representa el retardo

I_e representa la degradación introducido por los equipos de red. Básicamente este factor depende del codec y de las pérdidas de red

A representa el margen de seguridad

Simplifiquemos las cosas un poco

Para los impacientes que ya quieren comenzar a hacer cálculos matemáticos les diré que la recomendación ITU nos provee de una fórmula más simplificada para calcular el parámetro R, aunque obviamente menos precisa que la primera pues supone algunos valores por omisión. Esta fórmula es la siguiente.

$$R = 94,2 - I_d - I_e$$

Nota: Si somos observadores notaremos que el valor máximo que podría tomar R es de 94,2.

Ahora bien, como mencionamos antes los parámetros I_d e I_e representan el retardo y la pérdida de paquetes, pero esto no significa que vayamos a reemplazar estos valores por las mediciones de retardo en milisegundos y de pérdidas en porcentaje directamente en la fórmula. No, no; debemos antes adecuar estos valores y normalizarlos con ciertas fórmulas provistas por la recomendación.

El parámetro I_d representa el retardo y se calcula con la siguiente fórmula.

$$I_d = 0.024 \cdot d + 0.11 \cdot (d - 177.3) \cdot H(d - 177.3)$$

Donde d es el retardo en milisegundos y $H()$ es la función de Heavyside.

$H(x) = 0$ para $x < 0$ y 1 para $x \geq 0$.

Al analizar la fórmula de I_d notaremos a primera vista que hay un valor de inflexión en los 177.3 milisegundos y a valores mayores que éste la calidad de voz se ve afectada a un ritmo más severo. Esto es un dato interesante ya que nos permite bosquejar una regla empírica y decir que en las instalaciones siempre deberíamos mantener el retardo de la red debajo de los 170 milisegundos (redondeando un poco 177.3 milisegundos para podernos aprender ese valor).

Finalmente hay que calcular I_e para poder reemplazar este valor en la fórmula y calcular el dichoso R . Como dijimos antes I_e depende de algunos factores pero básicamente de las pérdidas de paquetes y del *codec* usado. Para mantener las cosas relativamente simples usaremos un gráfico con el cual podremos determinar de manera aproximada el valor de I_e dependiendo de, precisamente, el porcentaje de pérdida de paquetes y el *codec* usado. La tabla es la siguiente.

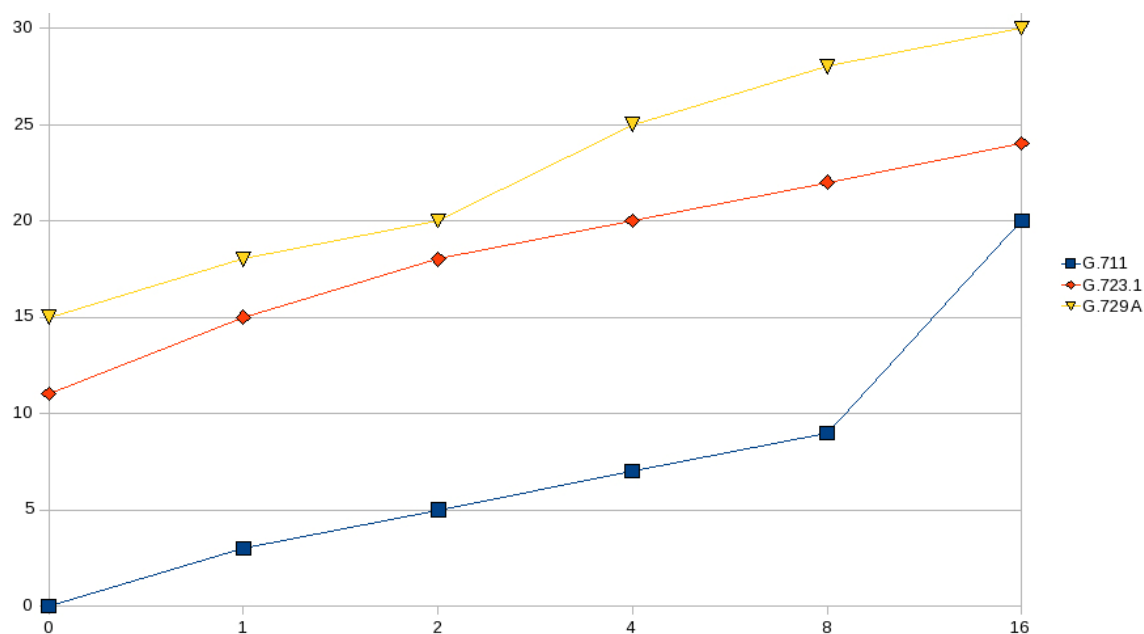


Gráfico I_e vs Pérdida de paquetes

Este gráfico fue generado a partir de la siguiente tabla de valores aproximados. El eje horizontal representa la pérdida de paquetes y el eje vertical representa el parámetro I_e .

| Packet Loss % | G.711 | G.723.1 | G.729A |
|---------------|-------|---------|--------|
| 0 | 0 | 11 | 15 |

| | | | |
|----|----|----|----|
| 1 | 3 | 15 | 18 |
| 2 | 5 | 18 | 20 |
| 4 | 7 | 20 | 25 |
| 8 | 9 | 22 | 28 |
| 16 | 20 | 24 | 30 |

Podemos observar que mientras más compresión involucre el uso de determinado *codec*, mayor es la contribución al parámetro *Ie* y por lo tanto menor la calidad de voz. Incluso podemos observar que con cero pérdidas ya algunos *codecs* merman la calidad de voz.

Traduciendo R a la escala MOS

Ahora que ya tenemos un valor de R vamos a explicar cómo traducirlo a la más conocida escala MOS.

La fórmula es la siguiente.

$$\begin{aligned}
 MOS &= 1 & R < 0 \\
 MOS &= 1 + 0.035 \cdot R + 7 \cdot R \cdot (R - 60) \cdot (100 - R) \cdot 10^{-6} & 0 < R < 100 \\
 MOS &= 4.5 & R > 100
 \end{aligned}$$

Ahora ya tenemos una manera práctica y objetiva de medir la calidad de voz de una instalación de voz sobre IP y de esta manera hacer ajustes correspondientes para mejorar la satisfacción de los usuarios telefónicos.

Ejemplo práctico

Veamos un caso de la vida real. Supongamos que hemos realizado algunas mediciones en un punto de la red y queremos determinar la calidad de voz en dicho punto en la escala MOS conociendo que usamos el *codec* G.711. Tenemos que el retardo es de 143 ms y la pérdida de paquetes llega al 7%. Cuál sería la medida de calidad de voz según la escala de MOS?

No entraré en detalles del desarrollo del problema aquí pero en algún momento llegaremos a la siguiente expresión.

$$R = 94.2 - 0.024 \cdot 143 - 7.2$$

$$R = 83.57$$

Traduciendo R a la escala MOS tendríamos:

$$MOS = 4.15$$

Lo cual no es un resultado de todo malo pues todavía nos encontramos en el rango de usuarios “Satisfechos”.

13.4 Canceladores de eco

Cómo funciona un cancelador de eco?

Es muy importante saber cómo funcionan los canceladores de eco para entender cómo sacarles el máximo provecho y quién sabe, para tener las bases para diseñar el nuestro propio o modificar alguno existente. A fin de cuentas estamos hablando de código abierto verdad?

Un cancelador de eco parte de un principio lógico muy sencillo. Si se refleja una parte de la señal de ida en la de regreso, entonces para eliminar el eco debería bastar con restar la señal de ida (en cierta proporción) a la señal de regreso.

Debido a que el eco es una señal reflejada que se encuentra atenuada, la clave está en estimar adecuadamente dicha atenuación. Por ejemplo, supongamos que la señal de eco es un 20% de la señal original. Entonces si podemos predecir que debemos restar a la señal de regreso un 20% de la señal de ida, habremos eliminado el eco completamente. Al menos en teoría claro está.

Esta “predicción” del factor de atenuación es una parte clave de todo algoritmo de cancelación de eco y recibe el nombre de algoritmo adaptativo o **filtro adaptativo** (*adaptive filter*) ya que debe adaptar el valor constantemente para obtener los mejores resultados.

En la siguiente figura supondremos que la atenuación de la señal reflejada está denotada por la letra α . La señal recibida es Rx, la señal transmitida es Tx y la señal transmitida con el componente de eco es Txe.

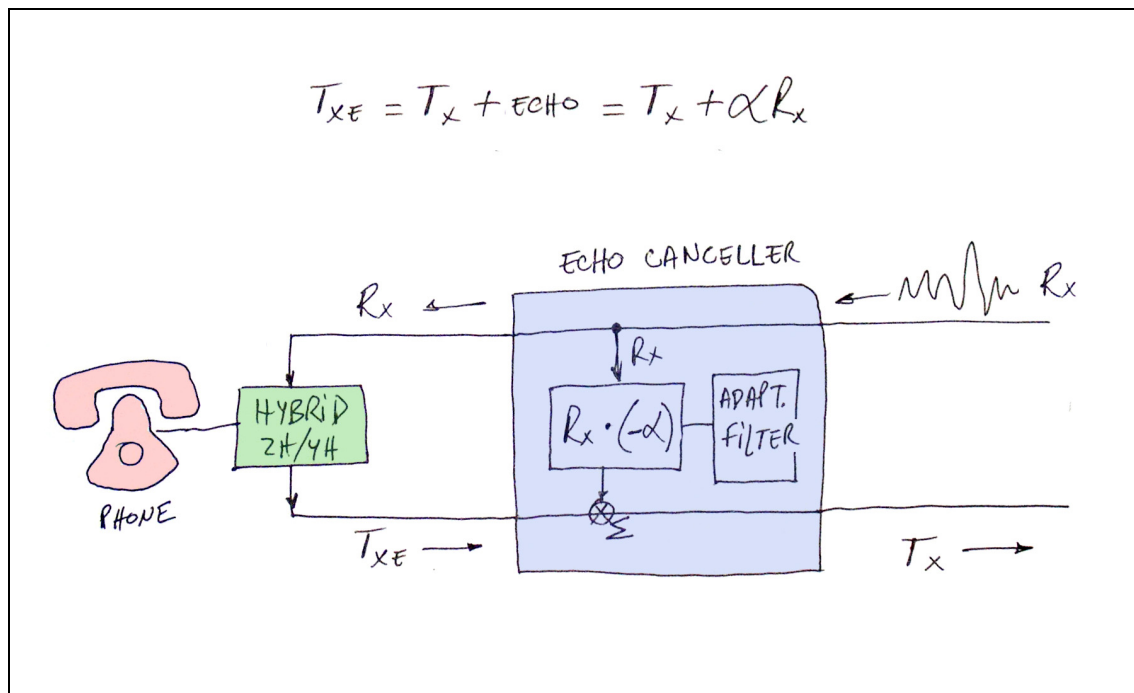


Diagrama básico del funcionamiento de un cancelador de eco

Sin embargo, no todo es tan sencillo como se explica aquí. El eco no es solo una señal reflejada atenuada sino que también puede estar desplazada en el tiempo con cierto retardo. Ese retardo también hay que predecirlo para poder restar la indeseable señal de eco. Predecir el retardo no es una tarea sencilla y por lo general los canceladores de eco tienen limitantes. Si el retardo es muy grande dejan de funcionar ya que es mucho trabajo revisar la señal en busca de retardo por grandes lapsos de tiempo.

***Nota:** Hay que hacer un pequeño paréntesis aquí y recordar que los canceladores de eco deben funcionar a tiempo real y por lo tanto no pueden perder el tiempo en cálculos muy tediosos ya que dejarían de ser veloces y retrasarían el envío de la voz.*

Valores típicos de retardo que un cancelador de eco puede manejar son 64 ms o 128 ms. Obviamente mientras menor sea el valor menor es el trabajo para el cancelador de eco (en términos de uso de CPU) pero mayor podría ser la imprecisión. Este tiempo máximo de retardo que un cancelador de eco puede tolerar se denomina comúnmente cola o *tail*.

El estándar G.168

Hace algunos años el ITU lanzó una especificación que define los parámetros de lo que debe ser un cancelador de eco. Podríamos decir que el documento es una recomendación. Por lo tanto, si un cancelador de eco está en concordancia con dicha recomendación se dice que es G.168 compatible (*compliant with G.168*). Sin embargo, G.168 es una especificación técnica compleja y no es cosa fácil determinar si un cancelador de eco es compatible o no. Existen empresas inclusive que se dedican al trabajo de revisar si un cancelador de eco es compatible o no con G.168.

No vamos a explicar aquí los detalles de G.168 pues sale del alcance del presente libro. Sin embargo, si alguien está interesado en detalles técnicos le recomendamos visitar el sitio del ITU (<http://www.itu.int>).

Un cancelador de eco para Asterisk que dice ser G.168 compatible es SoftEcho de OctWare.

14

Protocolos VoIP al descubierto

Within a few years a simple and inexpensive device, readily carried about, will enable one to receive on land or sea the principal news, to hear a speech, a lecture, a song or play of a musical instrument, conveyed from any other region of the globe.

-- Nikola Tesla, 1905

14.1 Protocolo SIP

El protocolo SIP (Session Initialization Protocol) es un protocolo de señalización (*application-layer control*) creado para administrar sesiones multimedia entre dos o más partes. Muchos se confunden pensando en que el protocolo SIP es quien transporta la voz propiamente dicha pero no es así, esta labor la realiza otro protocolo que veremos más adelante; de lo que sí se encarga SIP es de la comunicación necesaria para establecer una llamada, modificarla, así como para finalizarla.

El protocolo SIP fue diseñado con la premisa de la simplicidad en mente, se trata de un protocolo de texto con mensajes de comunicación sencillos. Inclusive SIP comparte algunos códigos de estatus con HTTP como el familiar “404: Not found”

Es un protocolo peer-to-peer. Es decir que prácticamente toda la lógica es almacenada en los *endpoints*.

Detalles del protocolo

- RTP es el portador de la voz y el video.
- SDP se usa para negociar las capacidades de los *endpoints*.
- Usa dos importantes protocolos como RTP/RTCP y SDP.
- La última versión del estándar es la RFC3261.
- Basado en texto, lo que nos permite entender los mensajes de una manera relativamente fácil.

Señalización SIP: Métodos y respuestas

Métodos

| Método | Descripción |
|----------|---|
| INVITE | Invita a un usuario a una llamada |
| ACK | Facilita el intercambio de mensajes confiables |
| BYE | Termina una conexión entre usuarios o declina una llamada |
| CANCEL | Termina un requerimiento o búsqueda por un usuario |
| OPTIONS | Solicita información acerca de capacidades del servidor SIP |
| REGISTER | Registra una ubicación de usuario |
| INFO | Usado para señalización en mitad de sesión. Para intercambiar información |

Respuestas

- 1xx Informational (e.g. 100 Trying, 180 Ringing)
- 2xx Successful (e.g. 200 OK, 202 Accepted)
- 3xx Redirection (e.g. 302 Moved Temporarily)
- 4xx Request Failure (e.g. 404 Not Found, 482 Loop Detected)
- 5xx Server Failure (e.g. 501 Not Implemented)
- 6xx Global Failure (e.g. 603 Decline)

SIP y Asterisk

Algunos detalles a tener en cuenta cuando se usa el protocolo SIP con Asterisk son los siguientes:

- A pesar de que SIP es independiente de la capa de transporte (puede ser usado con TCP, UDP, ATM, X.25, entre otros) en Asterisk su implementación está limitada a UDP
- Por omisión se usa el puerto 5060, pero este parámetro se puede modificar en el archivo `sip.conf`
- SIP adolece de problemas de *NAT*
- En Asterisk es posible hacer diagnóstico del protocolo SIP. Se verá esto más adelante.

14.2 Protocolo IAX

El protocolo IAX (*Inter-Asterisk eXchange*) es un protocolo de señalización creado por Mark Spencer, el mismo creador de Asterisk, con el objetivo de solucionar algunos problemas existentes con otros protocolos. El protocolo todavía no es un estándar pero pretende serlo a través de un proceso de estandarización en la IETF.

En esencia IAX presenta tres ventajas muy interesantes sobre otras alternativas como SIP.

- Consume menos ancho de banda
- Soluciona mejor problemas de NAT
- Pasa más fácilmente a través de *firewalls*

Si reflexionamos acerca de estas ventajas nos daremos cuenta rápidamente que resultan perfectas para troncalización entre dos servidores Elastix. En otras palabras, es recomendable el uso de IAX para interconectar dos o más servidores Elastix entre sí.

La versión actual del protocolo es la versión 2. La versión anterior ha quedado obsoleta por lo que es común ver el nombre IAX2 como sinónimo de IAX.

IAX es un protocolo binario, a diferencia de SIP que como recordaremos es un protocolo basado en texto. Esto es una ventaja desde el punto de vista desde ancho de banda puesto que en binario se desperdiciarán menos *bytes*.

IAX usa UDP y normalmente usa el puerto 4569. Lo interesante de IAX es que por un solo puerto transmite tanto la voz como la señalización y es esto lo que le permite resolver problemas de NAT y pasar a través de *firewalls* sin mayor inconveniente.

Además de esta característica el protocolo permite la troncalización de varios canales de audio en el mismo flujo de datos. Es decir que en un mismo datagrama se pueden enviar varias sesiones al mismo tiempo, lo que significa una reutilización de datagramas y por consiguiente un ahorro de ancho de banda.

Fases de una llamada IAX

Una llamada IAX tiene tres fases:

Establecimiento de la llamada

Para iniciar una llamada el equipo que la inicia (equipo A) le envía un mensaje NEW al equipo B y éste último responde con un mensaje ACCEPT. Luego de esto obviamente el equipo destino timbrará en espera de que el usuario conteste. Si esto sucede, el equipo B enviará un mensaje ANSWER al equipo A para notificar que el usuario contestó.

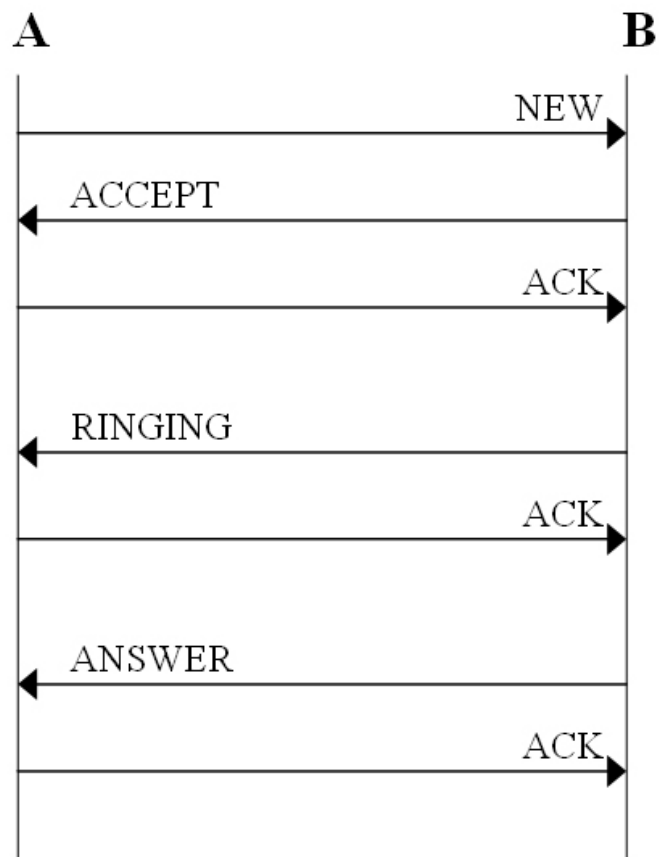
Nota: Se han omitido intencionalmente algunos mensajes ACK en la explicación de arriba para añadir claridad a la explicación.

Llamada en curso

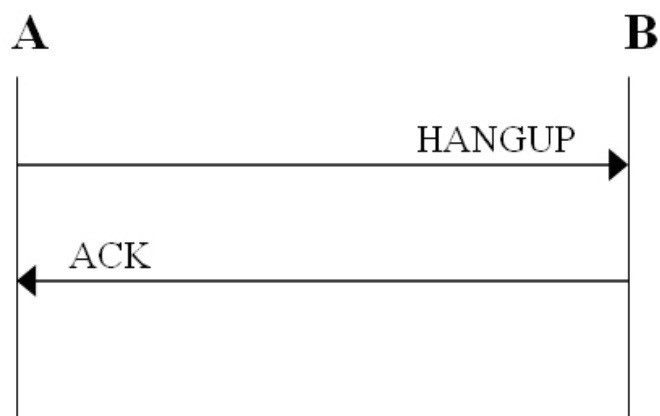
Si se contestó la llamada se inicia el intercambio de audio mediante unos paquetes llamados frames que estudiaremos más adelante. Estos frames se envían dentro del mismo flujo de comunicación que la señalización inicial.

Colgado

Para terminar la llamada cualquiera de las partes involucradas debe enviar un mensaje HANGUP.



Establecimiento de una llamada IAX entre los puntos A y B



Colgado de una llamada IAX

Frames IAX

Para intercambiar el audio entre los participantes de la llamada se utilizan dos tipos de *frames* llamados Full y Mini. También se les suelen llamar F y M.

Una conversación está compuesta en su mayoría por *frames* tipo Mini cuya virtud es (como su nombre lo sugiere) ser ligeros. Esto quiere decir que tienen una cabecera

pierde en el camino simplemente se rellenará ese espacio con un silencio. Lo que técnicamente se llama ruido confortable (*comfort noise*).

Es por esta necesidad de transmitir la información en tiempo real que resulta obvio que RTP sea un acrónimo de Real Time Protocol.

A pesar de encargarse de casi toda la labor de transportar la voz, RTP no está solo y tiene un protocolo de apoyo llamado RTCP. RTCP no es del todo indispensable pero proporciona valiosa ayuda al momento de transportar la voz de manera óptima pues proporciona estadísticas e información de control que le permiten a Asterisk o al otro extremo tomar decisiones para mejorar la transmisión en caso de ser posible. Por lo tanto, los paquetes RTCP se transmiten periódicamente para comunicar dicha información a los equipos de voz involucrados.

RTP se define en el documento RFC 3550 así que veamos qué dice al respecto:

RTP provides end-to-end network transport functions suitable for applications transmitting real-time data, such as audio, video or simulation data, over multicast or unicast network services. RTP does not address resource reservation and does not guarantee quality-of-service for real-time services. The data transport is augmented by a control protocol (RTCP) to allow monitoring of the data delivery in a manner scalable to large multicast networks, and to provide minimal control and identification functionality. RTP and RTCP are designed to be independent of the underlying transport and network layers. The protocol supports the use of RTP-level translators and mixers.

Para tener una mejor visión de lugar que ocupa un paquete RTP en el modelo TCP/IP veamos la siguiente figura.

Estructura de un paquete RTP

Un paquete RTP se compone de un encabezado y la data (o *payload*). En encabezado contiene alguna información interesante que explicaremos en breve, pero antes veamos cómo luce un encabezado RTP.

| Byte 1 | | | | Byte 2 | | Byte 3 | Byte 4 |
|--|---|---|----|--------|----|---------------------|--------|
| V=2 | P | X | CC | M | PT | Número de Secuencia | |
| Timestamp | | | | | | | |
| Synchronization Source (SSRC) | | | | | | | |
| Contributing Source (CSRC) | | | | | | | |
| Extensión (opcional dependiendo del bit X) | | | | | | | |
| Data... | | | | | | | |

Ahora sí pasemos a explicar qué información contiene un encabezado RTP.

- **V** es el número de versión. Este campo es de 2 bits de longitud y su valor contenido siempre es el número 2.
- **P** o *padding* es un bit que indica si hay relleno al final de la data o no. Si el bit está en uno quiere decir que si hay relleno. El relleno no es otra cosa que bytes adicionales al final del payload.
- **X** o extensión es un bit que indica si hay extensión del encabezado
- **CC** es un identificador de 4 bits que indica el conteo CSRC
- **M** o marcador de un bit
- **PT** o tipo de carga útil (*Payload Type*) es un identificador de 7 bits que nos indica el tipo de carga útil que contiene este paquete RTP. Ejemplos de tipos son G729, GSM, PCMU (G711 u-law), entre otros.
- **Número de Secuencia** (*sequence number*) es un número entero que identifica cada paquete del presente flujo de datos. Este es un identificador secuencial que se incrementa en uno con cada paquete transmitido. Ocupa 16 bits.
- **Timestamp** representa el instante de tiempo (en formato timestamp) en el que se comenzó a muestrear la data que está siendo transmitida en el payload. Ocupa 32 bits.
- **SSRC** identifica la fuente de sincronización ya que el mismo equipo puede estar “hablando” con diferentes fuentes de paquetes RTP. Es un número aleatorio de 32 bits por lo que hay la posibilidad (aunque la probabilidad es baja) de que este número se repita entre dos fuentes. Existen mecanismos para resolver este problema.
- **CSRC** es un número de 32 bits que identifica las fuentes contribuyentes para el payload

Luego de la cabecera vienen los datos.

15

Diagnóstico básico de problemas

Si hubiera preguntado a mis clientes qué necesitaban, habrían dicho un caballo mejor

-- Henry Ford, *fundador de Ford Motor Company*

15.1 Interpretando el *log*

Es muy importante entender cómo leer los archivos de *log* de Asterisk puesto que de allí podemos extraer información muy valiosa en nuestro proceso de diagnóstico.

Antes que nada es necesario conocer que tanto el nivel de detalle, así como el tipo de información que se escribe en los archivos de *log* son controlados desde el archivo `/etc/asterisk/logger.conf`. Este es un archivo de texto plano donde se definen los archivos donde se va a escribir información. Para cada uno de estos archivos se tiene que especificar también el nivel de detalle que se escribirá en ellos. Esto se hace en una sola línea, la sintaxis es la siguiente.

```
nombre_archivo => listado_de_niveles_separados_por_coma
```

Donde los niveles de *debugging* pueden ser los siguientes:

- debug
- notice
- warning
- error
- verbose

Nota: Hay un tipo especial de nombre de archivo que se encuentra reservado y que se llama "console". Representa la consola de Asterisk, o sea lo que se muestra en el Asterisk CLI.

Veamos ahora como ejemplo un fragmento del contenido del archivo `logger.conf` que se distribuye con Elastix.

```
; Directory for log files is configured in asterisk.conf
; option astlogdir
;
[logfiles]
;
; Format is "filename" and then "levels" of debugging to be
included:
;   debug
;   notice
;   warning
;   error
;   verbose
;
; Special filename "console" represents the system console
;
;debug => debug
;console => notice,warning,error
;console => notice,warning,error,debug
;messages => notice,warning,error
full => notice,warning,error,debug,verbose
```

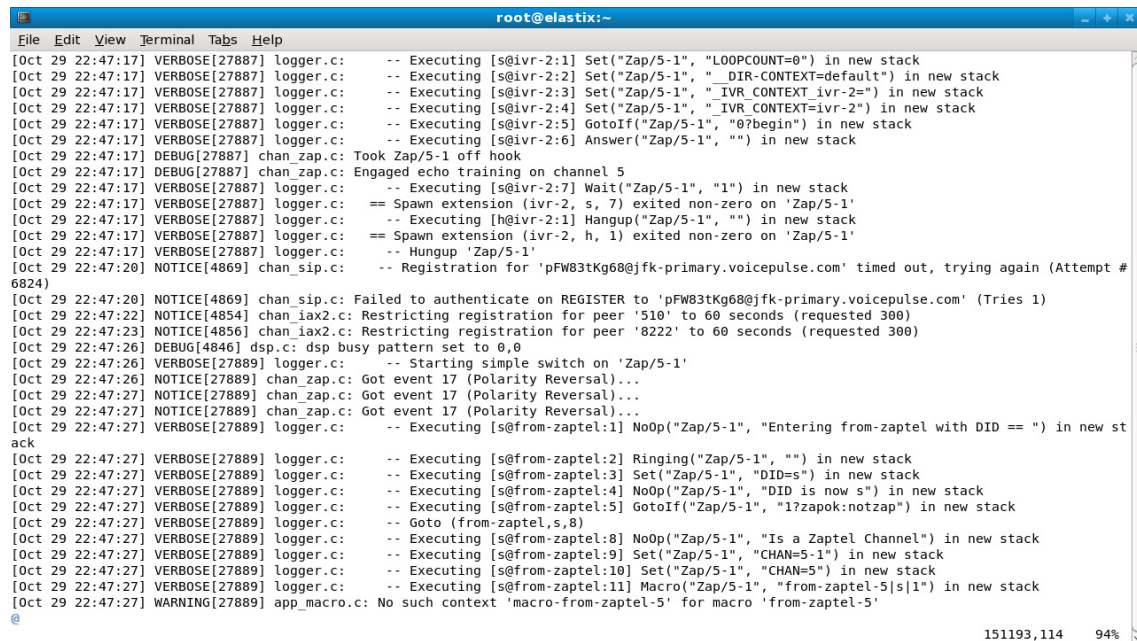
En el ejemplo anterior vemos algunas líneas comentadas y básicamente un solo archivo de *log* llamado `full`. Este, al igual que los demás archivos definidos de esta manera reside en la ruta `/var/log/asterisk/`

El archivo /var/log/asterisk/full

Como vimos en el último ejemplo, por omisión en Elastix tenemos un archivo de *log* llamado *full* donde Asterisk escribe información de su funcionamiento. Cada línea de este archivo tiene el siguiente formato.

[HORA FECHA] NIVEL[PID] NOMBRE_ARCHIVO: TEXTO

Veamos un ejemplo de cómo lucen algunas líneas del archivo de *log* de Asterisk.



```
File Edit View Terminal Tabs Help
[0ct 29 22:47:17] VERBOSE[27887] logger.c: -- Executing [s@ivr-2:1] Set("Zap/5-1", "LOOPCOUNT=0") in new stack
[0ct 29 22:47:17] VERBOSE[27887] logger.c: -- Executing [s@ivr-2:2] Set("Zap/5-1", " _DIR_CONTEXT=default") in new stack
[0ct 29 22:47:17] VERBOSE[27887] logger.c: -- Executing [s@ivr-2:3] Set("Zap/5-1", " IVR_CONTEXT=ivr-2=") in new stack
[0ct 29 22:47:17] VERBOSE[27887] logger.c: -- Executing [s@ivr-2:4] Set("Zap/5-1", " IVR_CONTEXT=ivr-2=") in new stack
[0ct 29 22:47:17] VERBOSE[27887] logger.c: -- Executing [s@ivr-2:5] GotoIf("Zap/5-1", "0?begin") in new stack
[0ct 29 22:47:17] VERBOSE[27887] logger.c: -- Executing [s@ivr-2:6] Answer("Zap/5-1", "") in new stack
[0ct 29 22:47:17] DEBUG[27887] chan_zap.c: Took Zap/5-1 off hook
[0ct 29 22:47:17] DEBUG[27887] chan_zap.c: Engaged echo training on channel 5
[0ct 29 22:47:17] VERBOSE[27887] logger.c: -- Executing [s@ivr-2:7] Wait("Zap/5-1", "1") in new stack
[0ct 29 22:47:17] VERBOSE[27887] logger.c: == Spawn extension (ivr-2, s, 7) exited non-zero on 'Zap/5-1'
[0ct 29 22:47:17] VERBOSE[27887] logger.c: -- Executing [h@ivr-2:1] Hangup("Zap/5-1", "") in new stack
[0ct 29 22:47:17] VERBOSE[27887] logger.c: == Spawn extension (ivr-2, h, 1) exited non-zero on 'Zap/5-1'
[0ct 29 22:47:17] VERBOSE[27887] logger.c: -- Hungup 'Zap/5-1'
[0ct 29 22:47:20] NOTICE[4869] chan_sip.c: -- Registration for 'pFW83tKg68@jfk-primary.voicepulse.com' timed out, trying again (Attempt #
6824)
[0ct 29 22:47:20] NOTICE[4869] chan_sip.c: Failed to authenticate on REGISTER to 'pFW83tKg68@jfk-primary.voicepulse.com' (Tries 1)
[0ct 29 22:47:22] NOTICE[4854] chan_iax2.c: Restricting registration for peer '510' to 60 seconds (requested 300)
[0ct 29 22:47:23] NOTICE[4856] chan_iax2.c: Restricting registration for peer '8222' to 60 seconds (requested 300)
[0ct 29 22:47:26] DEBUG[4846] dsp.c: dsp busy pattern set to 0,0
[0ct 29 22:47:26] VERBOSE[27889] logger.c: -- Starting simple switch on 'Zap/5-1'
[0ct 29 22:47:26] NOTICE[27889] chan_zap.c: Got event 17 (Polarity Reversal)...
[0ct 29 22:47:27] NOTICE[27889] chan_zap.c: Got event 17 (Polarity Reversal)...
[0ct 29 22:47:27] NOTICE[27889] chan_zap.c: Got event 17 (Polarity Reversal)...
[0ct 29 22:47:27] VERBOSE[27889] logger.c: -- Executing [s@from-zaptel:1] NoOp("Zap/5-1", "Entering from-zaptel with DID == ") in new st
ack
[0ct 29 22:47:27] VERBOSE[27889] logger.c: -- Executing [s@from-zaptel:2] Ringing("Zap/5-1", "") in new stack
[0ct 29 22:47:27] VERBOSE[27889] logger.c: -- Executing [s@from-zaptel:3] Set("Zap/5-1", "DID=s") in new stack
[0ct 29 22:47:27] VERBOSE[27889] logger.c: -- Executing [s@from-zaptel:4] NoOp("Zap/5-1", "DID is now s") in new stack
[0ct 29 22:47:27] VERBOSE[27889] logger.c: -- Executing [s@from-zaptel:5] GotoIf("Zap/5-1", "1?zapok:notzap") in new stack
[0ct 29 22:47:27] VERBOSE[27889] logger.c: -- Goto (from-zaptel,s,8)
[0ct 29 22:47:27] VERBOSE[27889] logger.c: -- Executing [s@from-zaptel:8] NoOp("Zap/5-1", "Is a Zaptel Channel") in new stack
[0ct 29 22:47:27] VERBOSE[27889] logger.c: -- Executing [s@from-zaptel:9] Set("Zap/5-1", "CHAN=5-1") in new stack
[0ct 29 22:47:27] VERBOSE[27889] logger.c: -- Executing [s@from-zaptel:10] Set("Zap/5-1", "CHAN=5") in new stack
[0ct 29 22:47:27] VERBOSE[27889] logger.c: -- Executing [s@from-zaptel:11] Macro("Zap/5-1", "from-zaptel-5[s]1") in new stack
[0ct 29 22:47:27] WARNING[27889] app_macro.c: No such context 'macro-from-zaptel-5' for macro 'from-zaptel-5'
151193,114 94%
```

Vista de archivo de log /var/log/asterisk/full

Observar el log interactivamente

Siempre es muy útil observar en tiempo real la actividad del servidor Elastix. Muchas veces nos veremos haciendo esto mientras realizamos o recibimos una llamada para tratar de averiguar qué falla. Podemos monitorear la salida en tiempo real con el comando *tail* de la siguiente manera.

```
tail -f /var/log/asterisk/full
```

Si queremos ver interactivamente solo los errores podemos añadir el siguiente filtro.

```
tail -f /var/log/asterisk/full | grep ERROR
```

Si queremos además ver los WARNINGS podemos modificar ligeramente el filtro de la siguiente manera.

```
tail -f /var/log/asterisk/full | grep -E "(ERROR|WARENING)"
```

***Nota:** Si queremos examinar todo el log en lugar de hacerlo interactivamente reemplazar el comando `tail -f` por el comando `cat`*

Qué información nos puede indicar el campo TEXTO?

Al mencionar la palabra TEXTO nos referimos a la información que viene al final de cada línea del *log* luego del nombre del archivo. Si se trata de una línea de ERROR encontraremos aquí una breve descripción del error ocurrido, si se trata de un WARNING también encontraremos un pequeño mensaje descriptivo. Sin embargo hay otro tipo de líneas con mensajes menos obvios y que simplemente describen el procesamiento de una llamada a través del plan de marcado. Estas líneas normalmente se encuentran macadas con nivel VERBOSE pues son simplemente informativas.

Examinar estas líneas de nivel VERBOSE o simplemente informativas también nos proveen de importante información que nos puede ser útil al momento de diagnosticar problemas.

En especial hay algunas de ellas que empiezan con la palabra "Excecuting" y a continuación mencionan en qué contexto y prioridad se encuentra Asterisk procesando el plan de marcado. Estas son las líneas de ejecución del plan de marcado. La sintaxis de estas líneas es la siguiente.

```
[TIME DATE] VERBOSE[PID] logger.c: --                               Excecuting
[extension@contexto:prioridad] MAS_TEXTO
```

Donde MAS_TEXTO puede contener información acerca de la Aplicación que fue ejecutada con los parámetros que le fueron pasados. Veamos un ejemplo de esto.

```
[Oct 28 17:52:34] VERBOSE[22034] logger.c: -- Executing
[s@macro-dial:7] Dial("Local/502@from-internal-e27c,2",
"SIP/502|21|trTWuWM(auto-blkvm)") in new stack
```

En este ejemplo vemos que cuando se escribió esta línea en el *log* nos encontrábamos dentro del contexto macro-dial (una macro, ya que empieza con la palabra macro), la extensión era la extensión s y la prioridad era la 7. También podemos observar que se está tratando de marcar mediante la aplicación Dial y podemos ver los parámetros que

se le pasaron como por ejemplo la cadena "trTWuWM(auto-blkvm)". En fin, podemos observar en todo caso que hay mucha información útil que podemos obtener de cada línea escrita en el *log*.

Número de proceso

Asterisk es un software multiproceso, es decir que puede crear varias instancias de si mismo que trabajen de manera coordinada entre si y de esta manera atender diferentes tareas al mismo tiempo. Cuando Asterisk escribe un línea en el *log* anota el número de proceso (o PID) que la escribió. Esto nos provee valiosa información puesto que podemos tener una visión más exacta del hilo de la llamada. Para hacer esto podemos filtrar por el número de proceso si es que lo conocemos. Por ejemplo.

```
cat /var/log/asterisk/full | grep 38484
```

Con el comando anterior estamos filtrando la salida por el proceso 38484. Por supuesto también podremos haber capturado alguna cadena de caracteres que coincida con el patrón 38484 como por ejemplo algún número de teléfono de algún usuario, pero estas líneas irrelevantes se las pueden eliminar manualmente.

Decíamos que gracias a este procedimiento podremos filtrar un poco más el hilo de una llamada. Esto es particularmente útil puesto que Asterisk escribe mucha información en el archivo de *log* y si nos encontramos en ambientes con muchas llamadas concurrentes ubicar información de una llamada específica en el *log* puede resultar un verdadero dolor de cabeza.

***Nota:** Algo importante a tener en cuenta es que una llamada puede tener más de un proceso involucrado.*

15.2 Diagnóstico a través del CLI

Desde el CLI podemos ver en tiempo real el funcionamiento de Asterisk. Es algo similar a hacer un "tail -f /var/log/asterisk/full" pero con algunas ventajas como por ejemplo que podemos ver el texto de la salida resaltado (coloreado) lo que mejora indudablemente la legibilidad y también está la posibilidad de interactuar simultáneamente con Asterisk al poder ejecutar algunos comandos desde la consola.

15.3 Sniffing

En el ámbito de *networking* el término *sniffing* significa capturar paquetes de red en su formato original para su posterior análisis. Existen herramientas especializadas para este fin y algunas cuestan una verdadera fortuna. Por suerte hay una de código abierto que hace un gran trabajo y se llama `Wireshark`.

Wireshark, anteriormente conocido como Ethereal, es un poderoso sniffer provisto con herramientas de análisis avanzadas, entre ellas algunas que nos permitirán analizar paquetes RTP y SIP.

Captura de llamadas

Antes que nada debemos pensar en qué tarjeta de red del servidor queremos capturar los paquetes de llamadas; al menos en los casos en los que tengamos más de una. Cuando ya estemos seguros la captura es sencilla, colocamos la laptop en la misma tarjeta

Finalizada la captura lo más probable es que junto con los paquetes de voz hayamos capturado otros paquetes que circulaban por la red y que no nos interesan como paquetes ICMP, requerimientos de DNSs, SMTP, etc. Con algo de práctica aprenderemos a filtrar estos paquetes irrelevantes, eso se lo dejamos de tarea al estudiante.

16

Control externo de Asterisk

Más que exigirlo, el respeto es algo que uno se gana

-- Anónimo

A veces editar el plan de marcado no es suficiente. Hace falta algún mecanismo más flexible para lograr lo que la empresa necesita. En otras ocasiones podemos inclusive poder controlar a Asterisk en tiempo real a través de alguna aplicación remota.

Para expandir el comportamiento de Asterisk, más allá del plan de marcado, existen dos interfases muy útiles llamadas AGI y AMI, que nos permitirán interactuar con Asterisk a un nivel mucho más poderoso que el logrado hasta ahora.

16.1 Asterisk AGI

AGI (Asterisk Gateway Interface) es una manera de interactuar con Asterisk desde un programa de línea de comandos. Este programa puede ser escrito en prácticamente cualquier lenguaje y es invocado por Asterisk desde el *dialplan*.

Al momento de la invocación de un programa AGI se le pasan ciertos parámetros y este a su vez responde con comandos AGI que son entendidos por Asterisk.

La utilidad de AGI es enorme y nos permite desarrollar funcionalidad extra para nuestra central telefónica. Un ejemplo muy común de esto es el desarrollo de aplicaciones de consulta telefónica a bases de datos.

Los programas o *scripts* AGI son muy populares y Elastix viene con algunos preinstalados en la carpeta `/var/lib/asterisk/agi-bin/`

Ejecución de *scripts* AGI

Los *scripts* AGI se comunican con Asterisk a través de STDIN, STDOUT y STDERR. Esto quiere decir que deberían poderse ejecutar desde la misma línea de comandos.

Algo muy importante a notar es que los *scripts* AGI deben ser ejecutables por el usuario `asterisk` así que es bueno asegurarse de que esto suceda.

```
# chown asterisk.asterisk script_prueba.agi
# chmod 755 script_prueba.agi
```

Luego de esto ya podemos ejecutar el *script* AGI desde el *dialplan* para que haga algo útil. Esto se lo hace gracias a la función `AGI()`. A continuación un ejemplo.

```
exten => 6789,1,Answer
exten => 6789,2,AGI(script_prueba.agi)
```

Como vemos en el ejemplo anterior le hemos dicho a Asterisk que ejecute el *script* llamado `script_prueba.agi`. Este *script* se ejecutará cuando el usuario digite la extensión 6789. Por omisión Asterisk asume que el *script* se encuentra en la ruta `/var/lib/asterisk/agi-bin/` así que si lo ubicamos en un lugar diferente habrá que especificar la ruta completa.

Parámetros enviados al *script* AGI

Como habíamos dicho antes, en el momento de la invocación del *script*, Asterisk le envía cierta información. A continuación un ejemplo de la información enviada cuando ejecutamos un *script* llamado `dialparties.agi`, un componente de freePBX.


```
agi_request: dialparties.agi
agi_channel: SIP/4444-08d6ab60
agi_language: en
agi_type: SIP
agi_uniqueid: 1222117320.271
agi_callerid: 4444
agi_calleridname: Edgar Landivar
agi_callingpres: 0
agi_callingani2: 0
agi_callington: 0
agi_callingtns: 0
agi_dnid: 216 6
agi_rdnis: unknown
agi_context: macro-dial
agi_extension: s
agi_priority: 3
agi_enhanced: 0.0
agi_accountcode:
```

Adicionalmente a la información que por omisión Asterisk le envía al *script* AGI, nosotros podríamos enviarle información adicional invocando la función AGI() de la siguiente manera.

```
exten => 6789,1,Answer
exten => 6789,2,AGI(script_prueba.agi|param1|param2|param3)
```

En el ejemplo anterior le estoy enviando 3 parámetros adicionales al *script*. Se pueden enviar tantos parámetros como sean necesarios.

Ya veremos más adelante cómo diferenciar estos dos grupos de parámetros desde la programación del *script*.

Comandos AGI

A continuación un listado de los comandos AGI disponibles en la última versión de Elastix 1.3. Este listado ha sido generado con el comando **agi show** desde el CLI.

| Comando | Descripción |
|----------------|---|
| answer | Answer channel |
| channel status | Returns status of the connected channel |

| | |
|---------------------|---|
| database del | Removes database key/value |
| database deltree | Removes database keytree/value |
| database get | Gets database value |
| database put | Adds/updates database value |
| exec | Executes a given Application |
| get data | Prompts for DTMF on a channel |
| get full variable | Evaluates a channel expression |
| get option | Stream file, prompt for DTMF, with timeout |
| get variable | Gets a channel variable |
| hangup | Hangup the current channel |
| noop | Does nothing |
| receive char | Receives one character from channels supporting it |
| receive text | Receives text from channels supporting it |
| record file | Records to a given file |
| say alpha | Says a given character string |
| say digits | Says a given digit string |
| say number | Says a given number |
| say phonetic | Says a given character string with phonetics |
| say date | Says a given date |
| say time | Says a given time |
| say datetime | Says a given time as specified by the format given |
| send image | Sends images to channels supporting it |
| send text | Sends text to channels supporting it |
| set autohangup | Autohangup channel in some time |
| set callerid | Sets callerid for the current channel |
| set context | Sets channel context |
| set extension | Changes channel extension |
| set music | Enable/Disable Music on hold generator |
| set priority | Set channel dialplan priority |
| set variable | Sets a channel variable |
| stream file | Sends audio file on channel |
| control stream file | Sends audio file on channel and allows the listener to control the stream |
| tdd mode | Toggles TDD mode (for the deaf) |
| verbose | Logs a message to the asterisk verbose log |

| | |
|----------------|---------------------------------|
| wait for digit | Waits for a digit to be pressed |
|----------------|---------------------------------|

Librerías AGI

A pesar de que un programa AGI puede ser escrito en prácticamente cualquier lenguaje, existen algunos lenguajes que ya cuentan con librerías para facilitar las labores de desarrollo de *scripts* AGI. Algunos de estos lenguajes son los siguientes.

- Pearl
- PHP
- Python
- Ruby
- C
- .NET

Para ver información actualizada de las librerías o *frameworks* disponibles para estos lenguajes recomendamos revisar el siguiente link: <http://www.voip-info.org/wiki-Asterisk+AGI>

En nuestro caso utilizaremos una librería para PHP llamada phpAGI con la que codificaremos un sencillo *script* AGI de ejemplo.

Ejemplo sencillo de *script* AGI

Aquí ilustraremos con un ejemplo muy sencillo lo que podemos lograr con un *script* AGI.

Lo que haremos será algo muy básico, un *script* que reproduzca un número cuando marquemos una extensión determinada. A continuación el código en PHP.

```
#!/usr/bin/php -q
<?php
    echo "SAY NUMBER 1234 \n";
    sleep(4);
?>
```

Este código lo podemos copiar en un archivo llamado `simple.agi` y para ser ordenados lo copiaremos en la ruta estándar `/var/lib/asterisk/agi-bin/`

Una vez copiado debemos estar seguros que el *script* pueda ser ejecutado por Asterisk así que sería buena idea ejecutar los siguientes comandos para asegurarnos de esto.

```
chmod 755 /var/lib/asterisk/agi-bin/simple.agi
chown asterisk.asterisk /var/lib/asterisk/agi-bin/simple.agi
```

Finalmente modifiquemos el plan de marcado para invocar nuestro *script* AGI cuando se marque una determinada extensión. Usaremos la extensión 6789 para este ejemplo.

```
exten => 6789,1,Answer
exten => 6789,2,AGI(simple.agi)
exten => 6789,3,Hangup
```

Este código anterior podría ir al final del archivo `/etc/asterisk/extensions_custom.conf`, ya que este archivo no es sobrescrito por freePBX.

Listo, ahora solo queda probar marcando la extensión 6789 y escuchar hablar a nuestro código!

16.2 Asterisk AMI

AMI (Asterisk Manager Interface por sus siglas en inglés) es una manera de comunicarse con Asterisk a través del protocolo IP. Digamos que es un concepto similar a AGI pero mientras que AGI está pensado para comunicaciones locales AMI puede ser ejecutado desde equipos remotos. Es esto último lo que lo convierte en una herramienta poderosa.

Gracias a AMI se pueden desarrollar poderosas aplicaciones cliente que pueden ejecutarse en otros computadores, liberando con esto de carga al servidor Asterisk. Un ejemplo de esto es el módulo para *call centers* que forma parte del proyecto Elastix. Este módulo de Elastix está codificado en gran parte usando AMI.

Debido a que a través de AMI equipos remotos están en la capacidad de controlar Asterisk se requiere que estos equipos se autenticuen. Estos usuarios, claves y permisos se definen en un archivo llamado `manager.conf`

A continuación el archivo `manager.conf` que se distribuye con Elastix

```

;
; Asterisk Call Management support
;
[general]
enabled = yes
port = 5038
bindaddr = 0.0.0.0

[admin]
secret = elastix456
deny=0.0.0.0/0.0.0.0
permit=127.0.0.1/255.255.255.0
read = system,call,log,verbose,command,agent,user
write = system,call,log,verbose,command,agent,user

#include manager_additional.conf
#include manager_custom.conf

```

Como podemos observar en el archivo anterior se define un usuario llamado `admin` con clave `elastix456` que puede conectarse desde el IP `127.0.0.1` con máscara `255.255.255.0`. Luego de esto se definen ciertos permisos para este usuario.

Probando desde *telnet*

No necesitamos de un programa sofisticado para probar si podemos interactuar con el AMI. Una manera sencilla de hacerlo es a través del programa `telnet` disponible en casi cualquier computador.

Veamos un ejemplo desde el propio servidor Elastix y suponiendo el archivo de configuración anterior.

Primero nos conectamos al propio servidor Elastix a través de una sesión `telnet`. Para esto ejecutamos lo siguiente desde la línea de comandos.

```

[root@elastix ~]# telnet 127.0.0.1 5038
Trying 127.0.0.1...
Connected to localhost.localdomain (127.0.0.1).
Escape character is '^]'.
Asterisk Call Manager/1.0

```

Como podemos ver el servidor nos ha contestado con un mensaje indicando que el AMI está escuchando y nos dice también que se trata de la versión 1.0 del protocolo.

A continuación nos autenticamos con el usuario y clave especificado en el archivo `manager.conf`

```
Action: Login
ActionID: 1
Username: admin
Secret: elastix456
```

Es importante conocer que para enviar un comando debemos ingresar dos retornos de carro. O sea que debemos dar ENTER un par de veces para enviar la petición de autenticación anterior. Luego de esto AMI nos contestará con un mensaje de éxito o error. En nuestro caso como ingresamos bien la clave nos contestará con éxito.

```
Response: Success
ActionID: 1
Message: Authentication accepted
```

Luego de esto comenzaremos a ver aparecer mensajes de registro que el AMI envía a nuestra sesión cada vez que ocurre un evento como por ejemplo cuando una llamada cuelga o si una extensión está timbrando.

A partir de aquí ya podemos interactuar con AMI a través de comandos, también llamados acciones (como veremos a continuación).

Tipo de paquetes AMI

En la comunicación por AMI se pueden identificar tres tipos de paquetes de información.

- **Acción:** Un comando enviado por el cliente al servicio AMI para que sea procesado.
- **Respuesta:** La respuesta de Asterisk a la Acción requerida por la aplicación cliente.
- **Evento:** Se trata de información de un evento asincrónico ocurrido en Asterisk. Por ejemplo, ingresó una llamada telefónica.

Para relacionar las respuestas con las acciones se puede incluir un ActionID en cada Acción. Asterisk devolverá este parámetro en la respuesta.

Comandos o acciones AMI

El siguiente listado de comandos AMI se obtuvo con el comando **show manager commands** desde el CLI de Elastix 1.3.

| Comando | Privilegios | Descripción |
|-----------------|-------------|---|
| AbsoluteTimeout | call,all | Set Absolute Timeout |
| AgentCallbackLo | agent,all | Sets an agent as logged in by callback |
| AgentLogoff | agent,all | Sets an agent as no longer logged in |
| Agents | agent,all | Lists agents and their status |
| ChangeMonitor | call,all | Change monitoring filename of a channel |
| Command | command,all | Execute Asterisk CLI Command |
| DBGet | system,all | Get DB Entry |
| DBPut | system,all | Put DB Entry |
| Events | <none> | Control Event Flow |
| ExtensionState | call,all | Check Extension Status |
| GetConfig | config,all | Retrieve configuration |
| Getvar | call,all | Gets a Channel Variable |
| Hangup | call,all | Hangup Channel |
| IAXnetstats | <none> | Show IAX Netstats |
| IAXpeers | <none> | List IAX Peers |
| ListCommands | <none> | List available manager commands |
| Logoff | <none> | Logoff Manager |
| MailboxCount | call,all | Check Mailbox Message Count |
| MailboxStatus | call,all | Check Mailbox |
| MeetmeMute | call,all | Mute a Meetme user |
| MeetmeUnmute | call,all | Unmute a Meetme user |
| Monitor | call,all | Monitor a channel |
| Originate | call,all | Originate Call |
| Park | call,all | Park a channel |
| ParkedCalls | <none> | List parked calls |
| PauseMonitor | call,all | Pause monitoring of a channel |

| | | |
|-----------------|------------|--|
| Ping | <none> | Keepalive command |
| PlayDTMF | call,all | Play DTMF signal on a specific channel |
| QueueAdd | agent,all | Add interface to queue |
| QueuePause | agent,all | Makes a queue member temporarily unavailable |
| QueueRemove | agent,all | Remove interface from queue |
| Queues | <none> | Queues |
| QueueStatus | <none> | Queue Status |
| Redirect | call,all | Redirect (transfer) a call |
| SetCDRUserField | call,all | Set the CDR UserField |
| Setvar | call,all | Set Channel Variable |
| SIPpeers | system,all | List SIP peers (text format) |
| SIPshowpeer | system,all | Show SIP peer (text format) |
| Status | call,all | Lists channel status |
| StopMonitor | call,all | Stop monitoring a channel |
| UnpauseMonitor | call,all | Unpause monitoring of a channel |
| UpdateConfig | config,all | Update basic configuration |
| UserEvent | user,all | Send an arbitrary event |
| WaitEvent | <none> | Wait for an event to occur |
| ZapDialOffhook | <none> | Dial over Zap channel while offhook |
| ZapDNDOff | <none> | Toggle Zap channel Do Not Disturb status OFF |
| ZapDNDon | <none> | Toggle Zap channel Do Not Disturb status ON |
| ZapHangup | <none> | Hangup Zap Channel |
| ZapRestart | <none> | Fully Restart zaptel channels (terminates calls) |
| ZapShowChannels | <none> | Show status zapata channels |
| ZapTransfer | <none> | Transfer Zap Channel |



GNU Free Documentation License

Version 1.3, 3 November 2008

Copyright (C) 2000, 2001, 2002, 2007, 2008 Free Software Foundation, Inc.

<<http://fsf.org/>>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals

providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been

arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

The "publisher" means any person or entity that distributes copies of the Document to the public.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

- A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
- B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.
- C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
- D. Preserve all the copyright notices of the Document.
- E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
- F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
- G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
- H. Include an unaltered copy of this License.
- I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.
- J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
- K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.
- L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
- M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.
- N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.
- O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at

your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties--for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See <http://www.gnu.org/copyleft/>.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Document.

11. RELICENSING

"Massive Multiauthor Collaboration Site" (or "MMC Site") means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A "Massive Multiauthor Collaboration" (or "MMC") contained in the site means any set of copyrightable works thus published on the MMC site.

"CC-BY-SA" means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

"Incorporate" means to publish or republish a Document, in whole or in part, as part of another Document.

An MMC is "eligible for relicensing" if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing.

B

Archivos de configuración de Asterisk distribuidos con Elastix

| Archivo | Descripción |
|-------------------------------|--|
| a2billing.conf | Archivo general de configuración de a2billing |
| additional_a2billing_iax.conf | Configuración de IAX para a2billing |
| additional_a2billing_sip.conf | Configuración de SIP para a2billing |
| adsi.conf | Archivo de configuración para ADSI (Interfase Análoga para servicios de <i>Display</i> por sus siglas en Inglés). Con ADSI se puede interactuar entre Asterisk y los <i>displays</i> de teléfonos análogos |
| adtranvofr.conf | Archivo de configuración que permite tener soporte de Voz sobre <i>Frame Relay</i> |
| agents.conf | Archivo de configuración que permite manejar los agentes de las colas. |
| alarmreceiver.conf | Sirve para configurar la aplicación AlarmReceiver() |
| alsa.conf | Configura el <i>driver</i> de sonido con los <i>drivers</i> de ALSA |
| amd.conf | Se utiliza para cuando se hace llamadas y |

| | |
|-------------------|--|
| | estas detectan una contestadora automática |
| applications.conf | Básicamente aquí se incluye plan de marcado. Los contextos aquí incluidos se les puede denominar aplicaciones y sus nombres empiezan con la cadena “app-“ |
| asterisk.conf | Archivo principal de configuración de Asterisk. Aquí se configuran los directorios de trabajo de Asterisk así como algunas opciones generales |
| cbmysql.conf | Archivo de configuración del módulo de Asterisk llamado CMySQL (<i>Conference Bridge MySQL</i>) usado para manejar las conferencias en Elastix |
| cdr_custom.conf | Este archivo define lo que se escribe en los CDRs de Asterisk |
| cdr_manager.conf | Este archivo controla si se deben enviar CDRs a través del AMI (<i>Asterisk Manager Interface</i>) |
| cdr_mysql.conf | Archivo de configuración del módulo cdr_mysql que permite registrar los CDRs en la base de datos MySQL |
| cdr_odbc.conf | Aquí se configura el módulo que permite registrar CDRs a través del <i>driver</i> ODBC de base de datos |
| cdr_pgsq.conf | Aquí se configura el módulo que permite registrar CDRs en la base de datos PostgreSQL |
| cdr_tds.conf | Con este archivo podemos configurar el módulo que permite registrar CDRs a través de los <i>drivers</i> de freeTDS. Por ejemplo, con este módulo podemos registrar CDRs en la base de datos MS-SQL |
| cdr.conf | El archivo principal de configuración de la reportación de CDRs |
| codecs.conf | En este archivo se configuran algunas opciones que controlan el funcionamiento de algunos <i>codecs</i> |
| dnsmgr.conf | Este archivo define si Asterisk debe realizar peticiones DNS regularmente y cada cuánto tiempo |
| dundi.conf | Archivo de configuración de DUNDI (<i>Distributed Universal Number Discovery</i>) |

| | |
|---|---|
| enum.conf | Archivo de configuración de ENUM |
| extconfig.conf | Permite configurar lo que se llama “configuración externa”. Es básicamente una funcionalidad que permite mapear cualquier archivo de configuración con una entidad externa como una base de datos |
| extensions_a2billing.conf | Plan de marcado adicional para la aplicación A2Billing |
| extensions_additional.conf | Archivo de configuración de Asterisk modificado por FreePBX donde no se debe editar a mano porque los cambios se perderán |
| extensions_custom.conf | Archivo donde se pueden crear contextos personalizados de Asterisk sin que FreePBX los sobre escriba |
| extensions.ael | Plan de marcado en formato AEL (<i>Asterisk Extension Language</i>). No se usa en Elastix. |
| extensions.conf | Archivo de configuración que incluye a extensions_additional.conf y a extensions_additional.conf |
| features_applicationmap_additional.conf | Archivo incluido desde el features.conf que contiene configuración de la sección applicationmap y que es sobrescrito por freePBX |
| features_applicationmap_custom.conf | Archivo incluido desde el features.conf que contiene configuración de la sección applicationmap y que puede ser sobrescrito por el usuario ya que no lo sobrescribe freePBX |
| features_featuremap_additional.conf | Archivo incluido desde el features.conf que contiene configuración de la sección featuremap y que es sobrescrito por freePBX |
| features_featuremap_custom.conf | Archivo incluido desde el features.conf que contiene configuración de la sección featuremap y que puede ser sobrescrito por el usuario ya que no lo sobrescribe freePBX |
| features_general_additional.conf | Archivo de configuración general de <i>features</i> que sobrescribe freePBX |

| | |
|------------------------------|--|
| features_general_custom.conf | Archivo de configuración general de <i>features</i> que puede ser escrito por el usuario ya que freePBX no lo sobrescribe |
| features.conf | Aquí se pueden configurar algunas características de Asterisk como parámetros de transferencias y captura de llamadas. |
| festival.conf | Arhcivo de configuración de Festival |
| followme.conf | Configuración de la funcionalidad <i>followme</i> |
| func_odbc.conf | Permite configurar parámetros de conexión con una base de datos a través del driver ODBC para poder hacer consultas en caliente a través del mismo plan de marcado |
| globals_custom.conf | Permite al usuario definir variables globales que no serán sobrescritas por freePBX. Este archivo es incluido desde el <code>extensions_additional.conf</code> |
| gtalk.conf | Archivo de configuración para que Asterisk funcione como cliente del servicio Google Talk |
| h323.conf | Archivo de configuración para protocolo H.323 según la implementación basada en el proyecto Open H.323. Existen otras implementaciones. |
| http.conf | Configuración de mini servidor Web embebido en Asterisk. Por omisión este servicio está apagado así que tener cuidado de no levantarlo en el mismo puerto que el servidor Apache que viene integrado en el Elastix |
| iax_additional.conf | Configuración de IAX generada por FreePBX no se debe editar |
| iax_custom.conf | Configuración de IAX que se puede editar ya que no la sobrescribe FreePBX |
| iax_general_additional.conf | Archivo complementario al <code>iax.conf</code> donde freePBX escribe configuración general. Esto es porque en la actualidad el archivo <code>iax.conf</code> ha quedado como un simple archivo de inclusión, o sea que incluye a otros archivos |
| iax_general_custom.conf | Similar al <code>iax_general_additional.conf</code> pero aquí puede escribir información el usuario ya que este archivo no es |

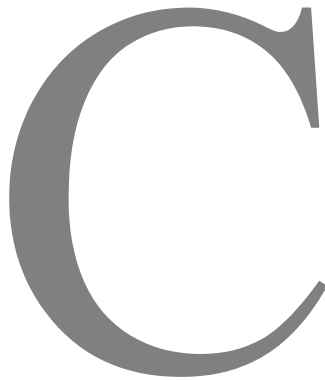
| | |
|-------------------------------|---|
| | sobrescrito por freePBX |
| iax_registrations_custom.conf | Similar al <code>iax_registrations.conf</code> pero este archivo no es sobrescrito por freePBX permitiendo almacenar configuración del usuario sin temor a que se borre |
| iax_registrations.conf | Archivo utilizado por freePBX para almacenar información de cadenas de registro con otras centrales o proveedores VoIP que usan protocolo IAX |
| iax.conf | Archivo principal de IAX que incluye a los otros |
| iaxprov.conf | Archivo utilizado par actualizar el <i>firmware</i> de dispositivos IAXy |
| indications.conf | Archivo para configurar la definición de tonos. Estos tonos pueden variar de un país a otro o de una compañía telefónica a otra. Algunos tonos analógicos comunes son el RING, BUSY, DIAL, entre otros. |
| jabber.conf | Archivo que permite configurar el soporte Jabber en Asterisk. Esto es porque Asterisk se puede conectar como cliente a un servidor Jabber (XMPP para decirlo más formal) |
| localprefixes.conf | Archivo que permite configurar reglas de marcado basadas en prefijos |
| logger.conf | En este archivo se puede configurar el <i>logging</i> de Asterisk. Ver la sección de diagnóstico para mayor información. |
| manager_additional.conf | Información de configuración de AMI que puede ser sobrescrita por freePBX |
| manager_custom.conf | Información de configuración de AMI para uso del usuario, o sea que este archivo no es sobrescrito por freePBX |
| manager.conf | Archivo de configuración del servicio de <i>Asterisk Manager Interface</i> . |
| meetme_additional.conf | Configuración de cuartos de conferencia escrita por freePBX |
| meetme.conf | Configuración de los cuartos de conferencia permanentes |
| mgcp.conf | Configuración de protocolo MGCP. |
| misdn.conf | Archivo de configuración del canal <code>chan_misdn</code> en Asterisk |
| modem.conf | Archivo de configuración del soporte para modems e ISDN usando los drivers de |

| | |
|-----------------------------|---|
| | ISDN4Linux o CAPI4Linux. |
| modules.conf | Este archivo le dice a Asterisk cuáles módulos se cargan y cuáles no. |
| musiconhold_additional.conf | Arvhivo para configuración de música en espera que sobrescribe freePBX |
| musiconhold_custom.conf | Definición de música en espera definida por el usuario. Este archivo no lo sobrescribe freePBX |
| musiconhold.conf | Aquí se configura la música en espera. Asterisk es muy flexible e inclusive soporta <i>streaming</i> de audio. |
| muted.conf | Permite enviar audio a ciertos canales y reducir el volumen de dicho audio mientras estos canales están activos. |
| osp.conf | Permite para configurar soporte para protocolo OSP (<i>Open Settlement Protocol</i>). Este protocolo nos permite intercambiar información (como <i>accounting</i> y uso) con proveedores que soporten dicho protocolo. |
| oss.conf | <p>Permite configurar canales tipo OSS (<i>Open Sound System</i>). Digamos que OSS es un <i>driver</i> de audio para simplificar las cosas y este canal nos permite configurar canales en la propia de tarjeta de sonido teniendo un softphone simplemente conectando un <i>headset</i>.</p> <p>Algo similar al soporte ALSA.</p> |
| parking_additional.inc | Archivo que almacena cierta información de lotes de parqueo. Este archivo se escribe automáticamente por el módulo Parkinglot |
| phone.conf | Permite configurar canales tipo phone que se basan en una implementación de canal para el proyecto Linux Telephony que soporta este tipo de dispositivos en Linux |
| phpagi.conf | Configura algunos parámetros básicos para el soporte de la librería PHPAGI |
| privacy.conf | Básicamente este archivo define el número de intentos que tiene un usuario para intentar ingresar un número de teléfono válido en la aplicación PrivacyManager() |

| | |
|--------------------------------|---|
| queues_additional.conf | Archivo que crea colas generado desde FreePBX, no se debe modificar manualmente porque se pueden perder los cambios |
| queues_custom_general.conf | Archivo de configuración general de colas escrito por el usuario, es decir que no lo sobrescribe freePBX |
| queues_custom.conf | Archivo de configuración de colas que puede ser modificado por los usuarios sin que FreePBX lo borre |
| queues_general_additional.conf | Archivo de configuración general de colas escrito por freePBX |
| queues_post_custom.conf | Actualmente freePBX no usa este archivo |
| queues.conf | Archivo general de configuración para colas. Si desea editar las colas en archivos de texto, se recomienda modificar queues_custom.conf |
| res_mysql.conf | Permite a Asterisk obtener la configuración de la base de datos MySQL. El concepto se denomina Asterisk Realtime |
| res_odbc.conf | Permite a Asterisk obtener configuración de bases de datos a través del driver ODBC |
| res_pgsq.conf | Permite a Asterisk obtener configuración de la base de datos PostgreSQL |
| res_snmp.conf | Permite configurar soporte SNMP en Asterisk |
| rpt.conf | Archivo de configuración de repetidora de radio. Esto le permite a Asterisk comunicarse vía VoIP utilizando tecnología de repetición de radio |
| rtp.conf | Configuración del protocolo RTP. Aquí se puede configurar el rango de puertos RTP que Asterisk utilizará |
| say.conf | Define ciertos parámetros de internacionalización |
| sip_additional.conf | Configuración de SIP generada por FreePBX y que no se debe editar sip_custom_post.conf |
| sip_custom.conf | Configuración de SIP que se puede modificar y no es sobre escrita por FreePBX |
| sip_general_additional.conf | Archivo complementario al sip.conf donde freePBX escribe configuración |

| | |
|--|--|
| | general. Esto es porque en la actualidad el archivo <code>sip.conf</code> ha quedado como un simple archivo de inclusión, o sea que incluye a otros archivos |
| <code>sip_general_custom.conf</code> | Similar al <code>sip_general_additional.conf</code> pero aquí puede escribir información el usuario ya que este archivo no es sobrescrito por freePBX |
| <code>sip_nat.conf</code> | Configuración para trabajar con SIP a través de NAT |
| <code>sip_notify.conf</code> | Utilizado para configurar el soporte que permite reiniciar algunos teléfonos SIP remotamente |
| <code>sip_registrations_custom.conf</code> | Similar al <code>sip_registrations.conf</code> pero este archivo no lo sobrescribe freePBX permitiendo almacenar configuración del usuario sin temor a que se borre |
| <code>sip_registrations.conf</code> | Archivo utilizado por freePBX para almacenar información de cadenas de registro con otras centrales o proveedores VoIP |
| <code>sip.conf</code> | Archivo de configuración de SIP que incluye a los otros |
| <code>skinny.conf</code> | Permite configurar el protocolo SCCP (<i>Skinny Client Control Protocol</i>) en Asterisk |
| <code>sla.conf</code> | Configuración de SLAs (<i>Shared Line Appearances</i>) |
| <code>smdi.conf</code> | Permite configurar soporte SMDI (<i>Station Message Desk Interface</i>) que le permite a Asterisk funcionar como un sistema de <i>voicemail</i> para centrales que soportan este protocolo |
| <code>udptl.conf</code> | Le permite a Asterisk soportar paquetes UDPTL. Estos paquetes son utilizados por el protocolo T.38 para fax |
| <code>unicall.conf</code> | Aquí se puede configurar el módulo <code>chan_unicall</code> |
| <code>users.conf</code> | Este archivo se utiliza para definir el concepto de “usuario” que puede tener asociado opcionalmente un teléfono (o extensión). Esto modela mejor el hecho de que un usuario puede cambiar de teléfono pero no necesariamente deja la compañía |

| | |
|------------------------|---|
| | (o la central telefónica) |
| vm_email.inc | Plantilla de email que se envía para notificar al usuario de la llegada de un nuevo correo de voz |
| vm_general.inc | Algunos parámetros generales relacionados con el <i>voicemail</i> |
| voicemail.conf | Archivo de configuración de los casilleros de voz |
| vpb.conf | Archivo de configuración de tarjetas VoiceTronix |
| zapata_additional.conf | Archivo que sobrescribe freePBX y que se incluye en el <code>zapata.conf</code> . Se utiliza básicamente cuando se crean extensiones tipo ZAP. |
| zapata-channels.conf | Archivo escrito por el <i>script</i> <code>genzaptelconf</code> y por el detector de hardware de Elastix para agregar configuración al <code>zapata.conf</code> |
| zapata.conf | Configuración de las tarjetas de telefonía que usan los drivers de <code>zaptel</code> . |



Lista de comandos de CLI

La siguiente lista de comandos ha sido generada en base al comando `help` sobre la versión de Asterisk 1.4.21.2.

| Comando | Descripción |
|---------------------------------|---|
| <code>!</code> | Execute a shell command |
| <code>abort halt</code> | Cancel a running halt |
| <code>ael debug contexts</code> | Enable AEL contexts debug (does nothing) |
| <code>ael debug macros</code> | Enable AEL macros debug (does nothing) |
| <code>ael debug read</code> | Enable AEL read debug (does nothing) |
| <code>ael debug tokens</code> | Enable AEL tokens debug (does nothing) |
| <code>ael nodebug</code> | Disable AEL debug messages |
| <code>ael reload</code> | Reload AEL configuration |
| <code>agent logoff</code> | Sets an agent offline |
| <code>agent show</code> | Show status of agents |
| <code>agent show online</code> | Show all online agents |
| <code>agi debug</code> | Enable AGI debugging |
| <code>agi debug off</code> | Disable AGI debugging |
| <code>agi dumhtml</code> | Dumps a list of agi commands in html format |
| <code>agi show</code> | List AGI commands or specific help |
| <code>cb mysql status</code> | Show connection status of CBMySQL |

| | |
|--|--|
| <code>cdr mysql status</code> | Show connection status of <code>cdr_mysql</code> |
| <code>cdr status</code> | Display the CDR status |
| <code>core clear profile</code> | Clear profiling info |
| <code>core set debug channel</code> | Enable/disable debugging on a channel |
| <code>core set debug</code> | Set level of debug chattiness |
| <code>core set debug off</code> | Turns off debug chattiness |
| <code>core set global</code> | Set global dialplan variable |
| <code>core set verbose</code> | Set level of verbosity |
| <code>core show applications</code> | Shows registered dialplan applications |
| <code>core show application</code> | Describe a specific dialplan application |
| <code>core show audio codecs</code> | Displays a list of audio codecs |
| <code>core show channels</code> | Display information on channels |
| <code>core show channel</code> | Display information on a specific channel |
| <code>core show channeltypes</code> | List available channel types |
| <code>core show channeltype</code> | Give more details on that channel type |
| <code>core show codecs</code> | Displays a list of codecs |
| <code>core show codec</code> | Shows a specific codec |
| <code>core show config mappings</code> | Display config mappings (file names to config engines) |
| <code>core show file formats</code> | Displays file formats |
| <code>core show file version</code> | List versions of files used to build Asterisk |
| <code>core show functions</code> | Shows registered dialplan functions |
| <code>core show function</code> | Describe a specific dialplan function |
| <code>core show globals</code> | Show global dialplan variables |
| <code>core show hints</code> | Show dialplan hints |
| <code>core show image codecs</code> | Displays a list of image codecs |
| <code>core show image formats</code> | Displays image formats |
| <code>core show license</code> | Show the license(s) for this copy of Asterisk |
| <code>core show profile</code> | Display profiling info |
| <code>core show switches</code> | Show alternative switches |
| <code>core show threads</code> | Show running threads |
| <code>core show translation</code> | Display translation matrix |
| <code>core show uptime</code> | Show uptime information |
| <code>core show version</code> | Display version info |
| <code>core show video codecs</code> | Displays a list of video codecs |
| <code>core show warranty</code> | Show the warranty (if any) for this copy of Asterisk |

| | |
|---------------------------|--|
| database del | Removes database key/value |
| database deltree | Removes database keytree/values |
| database get | Gets database value |
| database put | Adds/updates database value |
| database show | Shows database contents |
| database showkey | Shows database contents |
| dialplan add extension | Add new extension into context |
| dialplan add ignorepat | Add new ignore pattern |
| dialplan add include | Include context in other context |
| dialplan reload | Reload extensions and *only* extensions |
| dialplan remove extension | Remove a specified extension |
| dialplan remove ignorepat | Remove ignore pattern from context |
| dialplan remove include | Remove a specified include from context |
| dialplan show | Show dialplan |
| dnsmgr reload | Reloads the DNS manager configuration |
| dnsmgr status | Display the DNS manager status |
| dundi debug | Enable DUNDi debugging |
| dundi flush | Flush DUNDi cache |
| dundi lookup | Lookup a number in DUNDi |
| dundi no debug | Disable DUNDi debugging |
| dundi no store history | Disable DUNDi historic records |
| dundi precache | Precache a number in DUNDi |
| dundi query | Query a DUNDi EID |
| dundi show entityid | Display Global Entity ID |
| dundi show mappings | Show DUNDi mappings |
| dundi show peers | Show defined DUNDi peers |
| dundi show peer | Show info on a specific DUNDi peer |
| dundi show precache | Show DUNDi precache |
| dundi show requests | Show DUNDi requests |
| dundi show trans | Show active DUNDi transactions |
| dundi store history | Enable DUNDi historic records |
| feature show | Lists configured features |
| file convert | Convert audio file |
| group show channels | Display active channels with group(s) |
| help | Display help list, or specific help on a command |
| http show status | Display HTTP server status |
| iax2 provision | Provision an IAX device |

| | |
|---------------------------------------|--|
| <code>iax2 prune realtime</code> | Prune a cached realtime lookup |
| <code>iax2 reload</code> | Reload IAX configuration |
| <code>iax2 set debug</code> | Enable IAX debugging |
| <code>iax2 set debug jb</code> | Enable IAX jitterbuffer debugging |
| <code>iax2 set debug jb off</code> | Disable IAX jitterbuffer debugging |
| <code>iax2 set debug off</code> | Disable IAX debugging |
| <code>iax2 set debug trunk</code> | Enable IAX trunk debugging |
| <code>iax2 set debug trunk off</code> | Disable IAX trunk debugging |
| <code>iax2 show cache</code> | Display IAX cached dialplan |
| <code>iax2 show channels</code> | List active IAX channels |
| <code>iax2 show firmware</code> | List available IAX firmwares |
| <code>iax2 show netstats</code> | List active IAX channel netstats |
| <code>iax2 show peers</code> | List defined IAX peers |
| <code>iax2 show peer</code> | Show details on specific IAX peer |
| <code>iax2 show provisioning</code> | Display iax provisioning |
| <code>iax2 show registry</code> | Display IAX registration status |
| <code>iax2 show stats</code> | Display IAX statistics |
| <code>iax2 show threads</code> | Display IAX helper thread info |
| <code>iax2 show users</code> | List defined IAX users |
| <code>iax2 test loss_pct</code> | Set IAX2 incoming frame loss percentage |
| <code>indication add</code> | Add the given indication to the country |
| <code>indication remove</code> | Remove the given indication from the country |
| <code>indication show</code> | Display a list of all countries/indications |
| <code>keys init</code> | Initialize RSA key passcodes |
| <code>keys show</code> | Displays RSA key information |
| <code>local show channels</code> | List status of local channels |
| <code>logger mute</code> | Toggle logging output to a console |
| <code>logger reload</code> | Reopens the log files |
| <code>logger rotate</code> | Rotates and reopens the log files |
| <code>logger show channels</code> | List configured log channels |
| <code>manager show command</code> | Show a manager interface command |
| <code>manager show commands</code> | List manager interface commands |
| <code>manager show connected</code> | List connected manager interface users |
| <code>manager show eventq</code> | List manager interface queued events |
| <code>manager show users</code> | List configured manager users |
| <code>manager show user</code> | Display information on a specific manager user |

| | |
|------------------------|---|
| meetme | Execute a command on a conference or conferee |
| mgcp audit endpoint | Audit specified MGCP endpoint |
| mgcp reload | Reload MGCP configuration |
| mgcp set debug | Enable MGCP debugging |
| mgcp set debug off | Disable MGCP debugging |
| mgcp show endpoints | List defined MGCP endpoints |
| mixmonitor | Execute a MixMonitor command |
| module load | Load a module by name |
| module reload | Reload configuration |
| module show | List modules and info |
| module show like | List modules and info |
| module unload | Unload a module by name |
| moh reload | Music On Hold |
| moh show classes | List MOH classes |
| moh show files | List MOH file-based classes |
| no debug channel | |
| originate | Originate a call |
| pri debug span | Enables PRI debugging on a span |
| pri intense debug span | Enables REALLY INTENSE PRI debugging |
| pri no debug span | Disables PRI debugging on a span |
| pri set debug file | Sends PRI debug output to the specified file |
| pri show debug | Displays current PRI debug settings |
| pri show spans | Displays PRI Information |
| pri show span | Displays PRI Information |
| pri unset debug file | Ends PRI debug output to file |
| queue add member | Add a channel to a specified queue |
| queue remove member | Removes a channel from a specified queue |
| queue show | Show status of a specified queue |
| realtime load | Used to print out RealTime variables |
| realtime mysql status | Shows connection information for the MySQL RealTime driver |
| realtime pgsql status | Shows connection information for the Postgresql RealTime driver |
| realtime update | Used to update RealTime variables |
| restart gracefully | Restart Asterisk gracefully |

| | |
|-------------------------|---------------------------------------|
| restart now | Restart Asterisk immediately |
| restart when convenient | Restart Asterisk at empty call volume |
| rtcp debug ip | Enable RTCP debugging on IP |
| rtcp debug | Enable RTCP debugging |
| rtcp debug off | Disable RTCP debugging |
| rtcp stats | Enable RTCP stats |
| rtcp stats off | Disable RTCP stats |
| rtp debug ip | Enable RTP debugging on IP |
| rtp debug | Enable RTP debugging |
| rtp debug off | Disable RTP debugging |
| say load | set/show the say mode |
| show parkedcalls | Lists parked calls |
| show queues | |
| sip history | Enable SIP history |
| sip history off | Disable SIP history |
| sip notify | Send a notify packet to a SIP peer |
| sip prune realtime | Prune cached Realtime object(s) |
| sip prune realtime peer | Prune cached Realtime peer(s) |
| sip prune realtime user | Prune cached Realtime user(s) |
| sip reload | Reload SIP configuration |
| sip set debug | Enable SIP debugging |
| sip set debug ip | Enable SIP debugging on IP |
| sip set debug off | Disable SIP debugging |
| sip set debug peer | Enable SIP debugging on Peername |
| sip show channels | List active SIP channels |
| sip show channel | Show detailed SIP channel info |
| sip show domains | List our local SIP domains |
| sip show history | Show SIP dialog history |
| sip show inuse | List all inuse/limits |
| sip show objects | List all SIP object allocations |
| sip show peers | List defined SIP peers |
| sip show peer | Show details on specific SIP peer |
| sip show registry | List SIP registration status |
| sip show settings | Show SIP global settings |
| sip show subscriptions | List active SIP subscriptions |
| sip show users | List defined SIP users |
| sip show user | Show details on specific SIP user |
| skinny reset | Reset Skinny device(s) |

| | |
|-----------------------------------|---|
| <code>skinny set debug</code> | Enable Skinny debugging |
| <code>skinny set debug off</code> | Disable Skinny debugging |
| <code>skinny show devices</code> | List defined Skinny devices |
| <code>skinny show lines</code> | List defined Skinny lines per device |
| <code>sla show stations</code> | Show SLA Stations |
| <code>sla show trunks</code> | Show SLA Trunks |
| <code>soft hangup</code> | Request a hangup on a given channel |
| <code>stop gracefully</code> | Gracefully shut down Asterisk |
| <code>stop now</code> | Shut down Asterisk immediately |
| <code>stop when convenient</code> | Shut down Asterisk at empty call volume |
| <code>stun debug</code> | Enable STUN debugging |
| <code>stun debug off</code> | Disable STUN debugging |
| <code>transcoder show</code> | Display Zaptel transcoder utilization |
| <code>udptl debug</code> | Enable UDPTL debugging |
| <code>udptl debug ip</code> | Enable UDPTL debugging on IP |
| <code>udptl debug off</code> | Disable UDPTL debugging |
| <code>voicemail show users</code> | List defined voicemail boxes |
| <code>voicemail show zones</code> | List zone message formats |
| <code>zap destroy channel</code> | Destroy a channel |
| <code>zap restart</code> | Fully restart zaptel channels |
| <code>zap show cadences</code> | List cadences |
| <code>zap show channels</code> | Show active zapata channels |
| <code>zap show channel</code> | Show information on a channel |
| <code>zap show status</code> | Show all Zaptel cards status |

