

Comunicaciones Unificadas con Elastix

elastix_a_ritmo_de_merengue_rev_1.3.pdf

comunicaciones_unificadas_con_elastix_volumen_1_29mar2009.pdf

comunicaciones_unificadas_con_elastix_volumen_2_29mar2009.pdf

1 Introducción a la telefonía

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

- Mark Twain

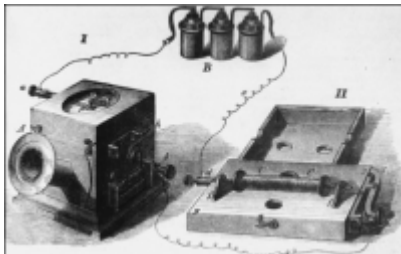
1.1 Breve historia de la telefonía

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 en tiempo real de alguien ubicado en un sitio remoto 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. Sin embargo, solo recién a inicios del siglo 19 fue cuando pareció 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

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



USA. 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. Este aparato estaba 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. 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 problemas con las patentes

Hasta este momento en la historia 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 documento expiró pocos años después. Meucci nunca pudo comercializar 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 "Mr. Watson, come here. I want to see you." ("Sr. Watson, venga. Necesito verlo.")

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 ganó Bell.

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.

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 aquellos tiempos. En 1886, ya existían más de 150,000 abonados telefónicos en los Estados Unidos.

A partir de ese entonces 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 éste tenía que solicitarle la llamada a una operadora. Esta persona era quien conectaba manualmente los cables para conmutar un punto con el otro. En

1891 se inventó un teléfono automático que permitía marcar directamente.

En un principio Bell fue 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 prestar este 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 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. Cabe mencionar que los abonados arrendaban el aparato telefónico a la compañía telefónica como parte del contrato.

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 sesenta 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.

En 1982, el Departamento de Justicia de los Estados Unidos gana un juicio a AT&T (nuevo nombre de Bell) en el cual esta última acuerda desmembrar la compañía en siete compañías regionales de telefonía conocidas como Regional Bell Operating Companies (RBOCs) o "Baby Bells". Todo esto a cambio de los derechos de ingresar en el mercado de las computadoras.

A partir de ésta y otras decisiones legales, los abonados pueden comprar sus propios aparatos telefónicos lo cual termina mermando el monopolio de AT&T. La compañía pierde mucha de su valuación y se ve en graves problemas financieros. Es irónico que en el 2005, AT&T terminase siendo adquirida por una de las Baby Bells, SBC Communications.

1.2 Los principios de la transmisión de voz

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). Esto es más rápido que un avión comercial. Pero esta rapidez no significa que usted se 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. Esto es más rápido de lo que podemos imaginar aunque para fines prácticos podemos decir que estas ondas se desplazan casi instantáneamente de un punto a otro. Otra cosa interesante es que podemos controlar la atenuación de estas ondas y hacerlas viajar 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 luego

transmitirlas 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.

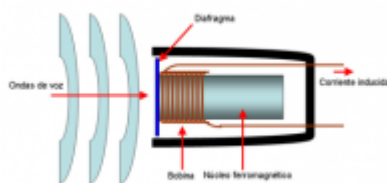
Rango de frecuencias de la voz humana

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 200Hz a 7kHz. ¡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 lo cual tiene sus ventajas pues facilita su transmisión como veremos más adelante. Por lo tanto los teléfonos comerciales solo transmiten un rango aproximado de 400Hz a 4kHz. Esto distorsiona en algo la voz aunque aún es posible 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 es el dispositivo que realiza 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.



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.

Ancho de banda y capacidad de información

Ancho de banda puede ser un término 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 se puede abreviar como 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 su 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, es más difícil que la comunicación se vea afectada por factores externos como el ruido eléctrico. Además nos provee de métodos para verificar periódicamente 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 (1s y 0s). Este proceso se denomina muestreo.

Teorema de Nyquist

En 1928 Henry Nyquist, un ingeniero Suizo que trabajaba para AT&T, resolvió el dilema del número mínimo de muestras necesarias para poder reconstruir una onda a su forma original.

El teorema propuesto dice que la frecuencia de muestreo debe ser como mínimo el doble de ancho de banda. Esto queda reflejado de mejor manera con la siguiente expresión.

$$f_m \geq 2 B_w$$

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

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

Más adelante veremos que precisamente esta frecuencia de muestreo de 8,000Hz es 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. El circuito es exclusivamente reservado para uso de una conversación y no puede ser usado por otros.

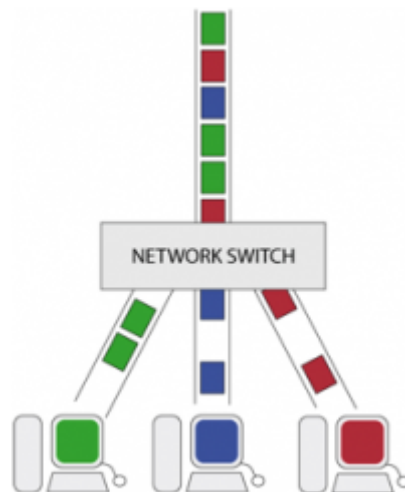
En cada circuito el retardo es constante, lo cual es una ventaja. Sin embargo, este tipo de redes es costoso debido al mismo hecho de que se necesita un circuito dedicado para las conversaciones de cada abonado.

Este tipo de redes es el tradicionalmente usado hoy en día 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

Las redes orientadas a paquetes (**packet switched**) son aquellas en las que se transmiten simultáneamente diferentes flujos de información en un mismo medio. Para hacer esto, se divide el tráfico de cada flujo de información en fragmentos o paquetes que se envían **intercaladamente**. Luego, en el destino los paquetes se **reensamblan** para reproducir el mensaje original.



Un ejemplo de este tipo de redes son las **redes IP** como es el caso del **Internet**, donde por una misma conexión pueden transmitirse distintos flujos de información. De esta manera podemos atender una 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 para llegar a su destino. Debido a esto, 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 quien hace posible que podamos comunicarnos con cualquier persona en nuestra ciudad o alrededor del mundo.

Originalmente fue una red analógica aunque actualmente es una red digital en su mayor parte. Es importante revisar los 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.

Actualmente, los circuitos analógicos están en decadencia pues las compañías telefónicas encontraron muchas ventajas en las comunicaciones digitales. Por esta razón, 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 se convierte a digital y es transmitida a un switch telefónico digital.

Los circuitos analógicos se asocian comúnmente con el término de “telefonía tradicional”.

Debido a que originalmente era más común que los teléfonos estuvieran ubicados en áreas rurales donde no llegaba la electricidad se decidió que la red telefónica proveyera cierto voltaje de alimentación. Es por esto que algunos modelos de teléfonos analógicos no necesitan conectarse a la alimentación eléctrica.

En todo caso la Oficina Central (OC) 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 tres 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 detalladamente 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, 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.
- **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.
- **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. Cabe resaltar que la conmutación hoy en día es completamente automática a diferencia del pasado cuando era trabajo de la operadora.
- **Ringado o Timbrado** Una vez que la OC encuentra al abonado destinatario tratará de timbrarlo (ringing). La señal de ring es una onda senoidal de 20 Hz y de 90 Voltios de amplitud. 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 senoidales 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 (busy) 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 (busy) 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.

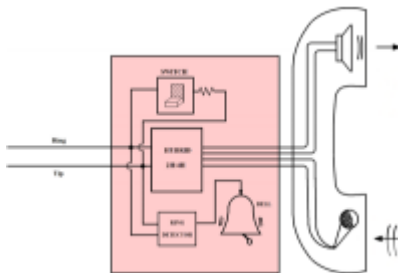
DTMFs

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.

	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

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. El teléfono analógico todavía es el tipo de teléfono más común en el planeta y la comprensión de su funcionamiento nos permitirá entender más adelante algunos conceptos clave como por ejemplo el eco.



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

La mayoría de los componentes se explican por si solos. Sin embargo algunos lectores se preguntarán de qué se trata el convertidor de 2 a 4 hilos también conocido como híbrido.

- **Convertidor de 2 a 4 hilos** Un componente importante de un teléfono analógico es el convertidor de 2 a 4 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 regreso ya que son dos los participantes en una conversación y solo existe un par de cables. 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.

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 multiplexados 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 trafica 2.048 Mbit/s.

Existen múltiplos mayores de T1 como T2, T3, y T4, los que se relacionan en la siguiente proporción.

$$1 \text{ T4} = 6 \text{ T3} = 42 \text{ T2} = 168 \text{ T1}$$

- **SONET y Circuitos Ópticos** 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 equivalente a 1 T3.

Existen múltiplos mayores de OC-1 como 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.

Estos protocolos se pueden agrupar en dos tipos llamados Señalización Asociada al Canal (CAS, por el acrónimo en Inglés) y Señalización de Canal Común (CCS, por el acrónimo en Inglés). 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. Es por esto que con CAS se reduce ligeramente el ancho de banda disponible 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.

Señalización Asociada al Canal (CAS)

El protocolo de tipo 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 a la pérdida de ese bit de información ya que es el bit menos significativo. Sin embargo, cuando transportamos data (al contrario de la voz) la pérdida de un bit no puede pasar desapercibida y la calidad de la transmisión se degrada de manera sensible.

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

Señalización de Canal Común (CCS)

El mejor ejemplo de protocolo de tipo CCS es el Integrated Services Digital Network (ISDN).

Este protocolo 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** 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 actualmente es utilizado en muchos países de este continente.
- **PRI: Primary Rate Interface** PRI es la opción ideal 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. Esto se debe a que la voz es muy sensible a retardos y problemas de transmisión por muy pequeños que estos sean.

Transmitir voz sobre protocolo IP es entonces 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.

La evolución de la calidad de las comunicaciones de voz sobre IP en los últimos años ha sido realmente sorprendente. Actualmente 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

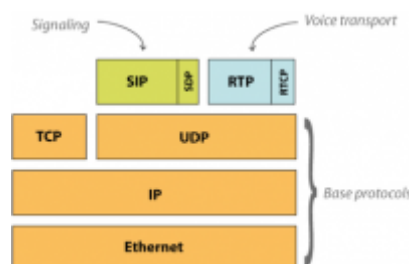
Hay muchos protocolos involucrados en la transmisión de voz sobre IP. Ya de por sí hay protocolos de red involucrados como el propio protocolo IP y otros protocolos de transporte como TCP o UDP. Encima de ellos se colocan los protocolos de señalización de voz. Como si esto fuera poco existen además muchas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



Clasificando los protocolos VoIP

Para simplificar las cosas podríamos clasificar a los protocolos utilizados en la VoIP en tres grupos: protocolos de señalización, de transporte de voz y de plataforma IP.

- **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. Estos protocolos se encuentran en la capa de sesión (capa 5) del modelo OSI. 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.
- **Protocolos de transporte de voz** No se debe confundir los protocolos de transporte de voz 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 su 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.



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 de la gran red de redes llamada Internet, que utiliza este protocolo para su enrutamiento. Por esta razón, antes de entrar en detalles acerca del protocolo IP emplearemos un par de párrafos para resumir los orígenes del Internet.

El Internet

El Internet tuvo su origen en la década del sesenta 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.

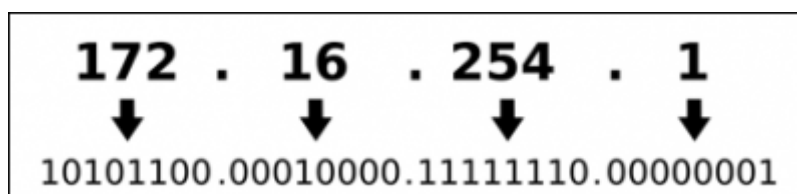
¿Qué es el protocolo IP?

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

Dirección IP

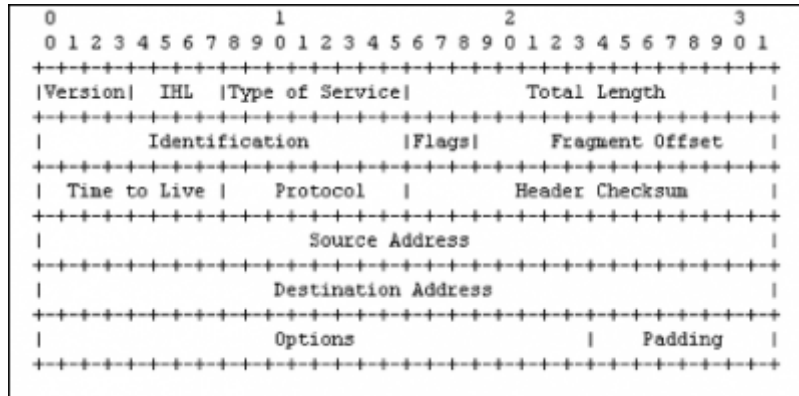
Una dirección IP es un número único que provee además información de cómo ubicar al equipo que la posee. Para que este número sea único existe una autoridad que controla la asignación de direcciones IP llamada IANA (Internet Assigned Numbers Authority).

Este número único es binario y 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 decimales puede tomar un valor de 0 a 255. Un ejemplo de dirección IP es 172.16.254.1



Paquete IP

Como habíamos dicho antes, el protocolo IP es un protocolo que divide la información en paquetes que envía a su destino. 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.



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 el destino para recuperar la información útil. A continuación una figura con una cabecera de paquete IP.

Direccionamiento IP

El direccionamiento o enrutamiento permite determinar la ruta óptima para que un paquete IP llegue 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 ruteadores y básicamente contienen tablas de rutas con información sobre cómo alcanzar otras redes. Por tanto, una vez que llega un paquete a un ruteador éste examina la dirección IP de 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 el número de bits que corresponden a la red y al host varían respectivamente. La solución es un parámetro adicional llamado máscara de red.

- **Máscara de red y cálculo de dirección de red** 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 como lo muestra el

diagrama a continuación.

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

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 parte de red en la máscara se completan con ceros. Convertamos ahora la dirección de red anterior a decimal. Veamos qué obtenemos:

```
10000010.00000101.00000101.00000000 à 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

- **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. Debido a esto es 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 TCP. Este 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, TCP 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 la transmisión de información en tiempo real como la voz. 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, mientras que 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. Imaginémos una flota de motocicletas que ofrece el servicio de transporte en una ciudad e imaginémos 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. La flota hará lo necesario para que nuestro cargamento llegue a su 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, 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 necesaria en los paquetes por lo que es un protocolo más rápido que TCP. Esto lo hace 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 (conocida como Analog-to-Digital Conversion, o ADC) pero resulta que nos encontramos en una red de paquetes así que debemos paquetizar esta información. Además existe la oportunidad de comprimir el resultado de la conversión ADC con algún algoritmo para no desperdiciar recursos de la red.

La búsqueda de un formato de compresión óptimo generó algunas alternativas de formatos de transmisión llamadas codecs.

Codecs

La palabra codec proviene de abreviar las palabras codificación y decodificación. 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.722, G.729, GSM, iLBC, entre otros.

Revisemos brevemente cuatro de ellos: G.711, G.722, G.729 y GSM.

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 un muestreo de 8 bits a 8kHz resultando en 64kbit/s. 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 viene habilitado por defecto en Elastix.

G.722

ofrece una tasa de muestreo de 14 bits a 16 kHz, es decir mucho mayor a G.711. En otras palabras recupera más información de la señal de audio y por lo tanto se dice que la calidad de la voz es superior. De hecho es el codec utilizado en muchos de los teléfonos VoIP marcados como de alta definición o HD (High Definition).

Algo muy interesante de G.722 es que pese a tener una tasa de muestreo mayor a la de G.711 y de que la resolución de cada muestra es también mayor a la de dicho codec, el consumo de ancho de banda es casi el mismo, es decir 64kbit/s. Esto es posible debido a que G.722 utiliza un algoritmo diferente.

Si analizamos rápidamente el ancho de banda de voz humana que este codec puede capturar (de acuerdo al teorema de Nyquist estudiado en el capítulo anterior), veremos que la tasa de muestreo de 16kHz nos permite digitalizar frecuencias de hasta 8kHz. Es decir una banda del doble de ancho que G.711, incluyendo frecuencias altas. Estas frecuencias altas contienen matices agudos que hacen que la voz sea más entendible y clara.

Con la reciente aparición en el mercado de teléfonos que dicen proveer audio de Alta Definición (HD) hemos visto un incremento en el uso de este codec. El lector podría pensar entonces que se trata de una tecnología nueva. Sin embargo, G.722 es un estándar de 1988.

G.729

Este también se trata de una recomendación ITU cuyas implementaciones ha sido históricamente licenciadas. Esto quiere decir que usualmente 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. Esto 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 (64kbit/s) notaremos que consume 8 veces menos ancho de banda, lo cual a simple vista es un ahorro de recursos significativo.

Para habilitar canales G.729 en Elastix se debe adquirir una licencia por cada canal.

GSM

Muchas personas correctamente asumen que el codec GSM tiene algo que ver con el estándar de comunicaciones celulares. 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. Sin embargo, la calidad de voz se deteriora en relación a otros codecs. Ya nos habremos fijado en esto cuando hemos hablado por un celular GSM alguna vez.

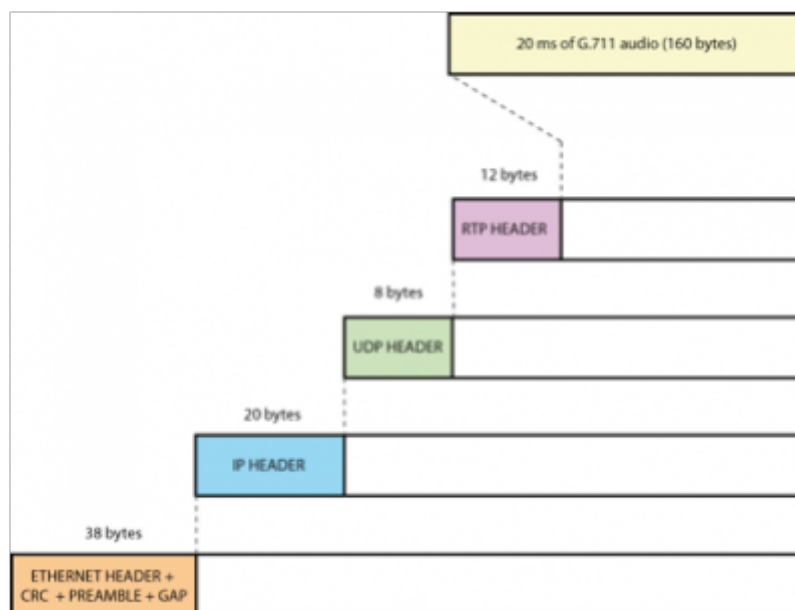
GSM comprime aproximadamente a 13kbit/s y viene habilitado por defecto en Elastix.

2.5 Sobrecarga de protocolos

Por lo anteriormente visto, para transportar la voz de un lugar a otro en una red de paquetes necesitamos la ayuda de algunos protocolos. Así también, ya nos habremos dado cuenta de que estos protocolos transmiten data adicional que ocupa ancho de banda adicional a la voz propiamente dicha. Algunos de ellos son Ethernet, IP, UDP y RTP. En resumen, esto hace que el ancho de banda real para transmitir la voz sea mayor al del codec.

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

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



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

Bytes transmitidos por cada 20ms de audio = $38 + 20 + 8 + 12 + 160 = 238$

bytes

Bits transmitidos por cada 20ms de audio = 238 bytes * 8 bits/byte = 1904 bits

Bits transmitidos cada segundo = 1904 bits/20ms * 1000 ms/seg . = 95,200 bits/s = ¡95.2Kbps!

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 tarea de los administradores de Elastix. Si usted ya conoce cómo administrar su Elastix desde la línea de comandos de Linux puede saltarse este capítulo e ir 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. Si al leer este capítulo en algún momento le parece que estamos abordando los temas de manera muy general, esta ha sido la intención.

Aunque este capítulo es un buen comienzo para el lector sin experiencia en Linux, le recomendamos capacitarse en su administración ya que sin duda le facilitará las tareas como administrador de Elastix.

3.2 Línea de comandos de Linux

Una manera muy cómoda de administrar Elastix es desde la línea de comandos de Linux. La línea de comandos de Linux nos permite explotar al máximo el potencial de nuestro servidor de comunicaciones unificadas.

Podemos ingresar a la línea de comandos directamente conectando un monitor y un teclado a nuestro equipo instalado con Elastix. Alternativamente, se puede conectar remotamente a través del protocolo SSH.

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

3.3 Comandos básicos

Comando	Descripción
<code>ls</code>	Enlista los archivos en el directorio actual
<code>ls -la</code>	Enlista los archivos y sus atributos
<code>cd <directorio></code>	Cambia de directorio
<code>cd ..</code>	Cambia al directorio padre
<code>cat <archivo></code>	Muestra el contenido de un archivo en la pantalla
<code>pwd</code>	Muestra la ruta al directorio actual
<code>cp <archivo1> <archivo2></code>	Copia un archivo o grupo de ellos
<code>mv <archivo1> <archivo2></code>	Mueve un archivo. También se lo usa para renombrar archivos
<code>rm <archivo></code>	Elimina un archivo
<code>tail -f <archivo></code>	Muestra las últimas líneas de un archivo en forma dinámica
<code>mkdir <directorio></code>	Crea un directorio
<code>tar -xvzf <archivo>.tar.gz</code>	Descomprime un archivo tar.gz
<code>top</code>	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
<code>find <directorio> -name <archivo></code>	Permite buscar archivos a partir de un directorio determinado
<code>vim <archivo></code>	Programa para ver y editar archivos de texto plano
<code>ifconfig</code>	Muestra información de las interfaces de red y permite modificarlas temporalmente
<code>ps -aux</code>	Muestra un reporte de los procesos que se están ejecutando en el servidor
<code>reboot</code>	Reinicia el equipo
<code>shutdown -h now</code>	Apaga el equipo

3.4 Sistema de archivos

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

Esta organización le otorga robustez e interoperabilidad a Linux pues los programas pueden predecir con mayor exactitud dónde se encuentran ciertos archivos para su correcto funcionamiento. Además, permite que los procesos escriban 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 hasta lo que hoy conocemos como Linux.

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 y se representa con una diagonal "/". Los directorios enlistados en la siguiente tabla residen todos en el directorio raíz.

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>	Contiene el kernel y otros archivos necesarios al momento de arranque
<code>dev</code>	Incluye archivos que representan dispositivos de hardware. Recordemos que en Linux prácticamente todo es un archivo
<code>etc</code>	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 <i>jorge</i>
<code>lib</code>	Contiene las librerías compartidas que requieren los programas. Algo análogo a los archivos <code>.dll</code> en Windows
<code>lost+found</code>	Cuando por alguna razón el sistema de archivos se corrompe y hay que repararlo, los resultados de esa reparación se guardan en este directorio
<code>mnt</code>	Este es un lugar común para montar otros sistemas de archivos o dispositivos. Por ejemplo una unidad de CD-ROM o una memoria USB
<code>opt</code>	Contiene paquetes de software que normalmente no son parte de la instalación original
<code>proc</code>	Este 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 DAHDI escriben ciertos archivos en la ruta <code>/proc/dahdi/</code>
<code>root</code>	Contiene archivos del usuario <i>root</i> ya que este no tiene uno del tipo <code>/home/root/</code>
<code>tmp</code>	Guarda archivos temporales
<code>usr</code>	Es uno de los directorios más importantes y contiene información para ser compartida con los usuarios del sistema. Es decir, aquí residen programas junto con la documentación necesaria y hasta librerías relacionadas. Usualmente, este directorio no contiene información que cambie constantemente
<code>var</code>	Contiene información en constante cambio como logs o colas de emails. Por ejemplo, aquí reside el log de Asterisk en la ruta <code>/var/log/asterisk/full</code>

Permisos

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

- Lectura
- Escritura
- Ejecución

Los permisos pueden ser aplicados al dueño del archivo, a un grupo de usuarios del sistema, o globalmente a todos los usuarios del sistema. Para cada una de estas entidades se puede establecer permisos diferentes.

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

En el ejemplo anterior, el usuario dueño tiene todos los permisos sobre el archivo mientras que los usuarios del grupo y otros usuarios globales del sistema solo tienen permisos de lectura y ejecución.

Existe sin embargo una notación más sencilla para expresar esta información. Esta notación compuesta de 10 caracteres simplifica el despliegue de la información de permisos y permite mostrarlos en una sola línea. Esto es muy útil cuando se quieren ver permisos de muchos archivos aunque toma tiempo acostumbrarse. Por ejemplo, la matriz de permisos del ejemplo anterior se puede expresar de la siguiente manera (hemos incluido la posición de cada permiso como un número del 0 al 9):

```
rwxr-xr-x  
0123456789
```

Donde,

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

Ahora veamos un ejemplo de un listado 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 dljl73.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
-rwxr-wr-- 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

```

Ejemplos de permisos de archivos

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 matriz de permisos de una forma más condensada llamada representación numérica u octal de permisos.

La idea es que los tres 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 esta forma toda la información de 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 o permisos globales.

Puede ser complicado entender al principio pero sin duda se trata de un formato más condensado. Veamos el siguiente ejemplo:

```

rwxr-xr-x   equivale a   755

```

Resulta que cada uno de los tres caracteres que representan un permiso para una entidad dada puede ser reemplazado por 1 o 0 de tal manera que obtenemos un número binario. Este número binario se traduce a su equivalente decimal (realmente deberíamos decir octal aquí) como se ve a continuación..

```

rwxr-xr-x

```

Transformado a binario constaría de los siguientes tríos (sin tomar en cuenta la primera posición 0 correspondiente al directorio):

```
111 101 101
```

y ahora en su equivalente decimal

```
755
```

En fin, para los que no les gusta calcular binario quizás la siguiente tabla sea útil.

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

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 (nótese la separación del usuario y grupo por un punto “.”).

```
# 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 recursiva con R.

3.5 RPMs y actualizaciones vía Yum

Los paquetes de software que se incluyen en Elastix vienen en un formato llamado RPM. Este 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 utilizando paquetes comprimidos usualmente con la extensión .rpm.

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, des-instalaciones, actualizaciones y otras tareas importantes.

A continuación se muestran algunas opciones de uso del comando rpm:

Acción a realizar	Comando
Instalación de RPMs	<code>rpm -ivh <paquete>.rpm</code>
Actualización de un RPM	<code>rpm -Uvh <paquete>.rpm</code>
Eliminación de un RPM	<code>rpm -e <paquete>.rpm</code>
Obtener información de un archivo RPM	<code>rpm -qpi <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. Sin embargo, 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. 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 servidor Elastix. ¡Vaya dolor de cabeza!

La solución es una utilidad llamada yum Este programa nos ayuda a instalar paquetes RPM desde repositorios y lo hace de una manera muy amigable.

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

A continuación mostramos algunos comandos yum útiles:

Comando	Descripción
<code>yum update</code>	Actualizar todo el sistema
<code>yum update <paquete></code>	Actualiza solo el paquete especificado y todas sus dependencias
<code>yum install <paquete></code>	Instala el paquete especificado y todas sus dependencias

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

4 Asterisk: Conceptos Esenciales

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

– Albert Einstein

4.1 ¿Qué es Asterisk?

Asterisk es un software de central telefónica (PBX) 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 electrónicas para conexión con la PSTN. Esto ha hecho que la oferta de centrales telefónicas basadas en Asterisk crezca rápidamente 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 central telefónica pero pronto se dio cuenta que estaba 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 funcional, decidió liberarlo bajo licencia GPL.

A pesar de lo que se pueda pensar, Mark no era un novato en el mundo del software libre. Él 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 software como tal.

Cabe resaltar que en un principio Mark se dio cuenta de que su software necesitaba interactuar con hardware telefónico. Fue entonces cuando se encontró con el proyecto Zaptel, un proyecto de código abierto creado por Jim Dixon. Zaptel tenía el objetivo de crear drivers abiertos para tarjetas telefónicas de computadora. A partir de allí Asterisk y Zaptel caminarían de la mano hasta el punto que en la actualidad los dos proyectos son mantenidos por Digium. En 2008 Digium decidió cambiar el nombre del proyecto que mantenían de Zaptel a DAHDI.

4.3 Funcionalidad provista por Asterisk

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

- Contestación automática de llamadas
- Transferencia de llamadas

- Opción de no molestar
- Parqueo de llamadas
- Contestación de llamada desde extensión remota
- Monitoreo y grabación de llamadas
- Voicemail
- Conferencias
- Reportación de llamadas
- Colas de atención
- Llamada en espera
- Identificador del llamante
- Bloqueo por llamante identificado
- Envío y recepción de fax
- Listado interactivo del directorio de extensiones
- Interactive voice response (IVR)
- Música en espera
- Manejo de comportamiento por horarios
- Follow me

Contestación automática de llamadas

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. Esta funcionalidad ha sido diseñada para empresas que reciben altos flujos de llamadas y que no poseen el personal necesario para manejar el tráfico de llamadas entrantes.

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

Transferencia de llamadas

En Asterisk existen dos métodos para transferir llamadas: transferencia atendida y transferencia ciega. En la transferencia atendida se llama primero al destinatario para avisarle que se le transferirá una llamada. En la transferencia ciega se transfiere simplemente la llamada sin conversar con el destinatario previamente.

Opción de no molestar

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

Parqueo de llamadas

El parqueo de llamadas permite al usuario que recibe una llamada enviarla a un espacio de parqueo,

para volver a atenderla desde otra extensión. Este tipo de acción es útil en muchas situaciones como cuando un técnico recibe una llamada a su escritorio la cual requiere que él se ubique de frente a un servidor localizado tres pisos más arriba.. En este caso, la opción es enviar la llamada a un cuarto temporal o espacio de parqueo, donde mientras tanto el llamante escuchará la música en espera configurada. El técnico puede retomar la llamada una vez que llegue al servidor.

Contestación de llamada desde extensión remota

Esta característica permite atrapar una llamada que se encuentra timbrando en una extensión que no es la suya. También se la conoce como call pickup, debido a su nombre en inglés.

Monitoreo y grabación de llamadas

Asterisk permite monitorear y grabar llamadas. El monitoreo se suele usar para fines de control de calidad 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 permita escuchar en línea la conversación sostenida por una extensión cualquiera.

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.

Voicemail (Buzón de mensajes de voz)

El buzón de mensajes de voz es una aplicación que permite guardar y escuchar mensajes dejados por llamantes que no pudieron ser atendidos.

Para acceder al buzón de mensajes de voz hay que digitar un código predeterminado. Asterisk nos preguntará por nuestra clave y a continuación accederemos a los mensajes disponibles.

Conferencias

Una conferencia no es más que una llamada telefónica donde pueden conversar varios participantes a la vez. Para acceder a esta funcionalidad se crean cuartos de conferencia. Cada cuarto se encuentra asociado a un número como si se tratase de una extensión. Entonces, para acceder a una conferencia basta con marcar el número correspondiente al cuarto de conferencia deseado.

Es posible configurar algunas características interesantes a un cuarto de conferencias determinado, como por ejemplo: limitar el número de participantes, solicitar una contraseña al ingreso o moderar el cuarto de conferencias.

Reportación de llamadas

Asterisk genera Call Detail Records (CDRs) 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, o por qué puerto o línea telefónica se realizó la llamada.

Colas de atención

Esta característica permite que un gran número de llamantes puedan permanecer en espera hasta que un representante o recurso esté disponible para dar asistencia. Esto asegura que los llamantes no terminarán obligatoriamente en el voicemail, lo cual puede resultar molesto. Es decir, con las colas de atención 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 la compañía, ofertas especiales, anuncios informativos ó simplemente 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. Para alertar a la persona que una segunda llamada está ingresando se le reproduce un sonido característico. En algunos escenarios este sonido no es deseable, por ejemplo en las conexiones telefónicas para transferencia de datos (Ej.: fax). En estos casos se opta por deshabilitar esta característica ya que provoca errores en la transmisión del fax.

Identificador del 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 de 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 ID.

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 incluye 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 telefónica 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.

Listado interactivo del directorio de extensiones

Asterisk cuenta con todo el directorio telefónico del personal de la empresa (nombre, apellido y extensión) en su base de datos. Esta facilidad permite 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 a quien 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 mejoran la forma en la cual un sistema telefónico acepta y distribuye sus llamadas. Con un menú de IVR es posible distribuir las llamadas entrantes de una manera precisa. Esto ayudará a incrementar el desempeño de los empleados y la satisfacción de los 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 en ciertos casos. Mediante el uso de esta característica se pueden crear menús de activación por dígitos ó por comandos de voz para ejecutar inclusive consultas a otros sistemas de cómputo como bases de datos.

Música en espera

Asterisk permite utilizar cualquier archivo en formato mp3 o wav como música de fondo. Ésta será reproducida cuando algún llamante externo o interno se encuentra esperando a que un representante de la compañía lo atienda. Estos archivos pueden ser inclusive agrupados por categorías para reproducir diferentes melodías a diferentes grupos de llamantes.

Manejo de comportamiento por horarios

Es posible configurar Asterisk para que tenga un tratamiento diferente para con las llamadas (tanto entrantes como salientes) dependiendo de la hora del día, día del mes o mes del año. El ejemplo más básico es cuando se quiere que las llamadas entrantes recibidas fuera de horarios de oficina reciban un mensaje pregrabado sugiriendo al llamar nuevamente al siguiente día.

Follow me

Esta característica, conocida en español como sígueme, permite predefinir una secuencia de llamadas simultáneas o en secuencia para poder ubicar a un usuario. Por ejemplo, se podría definir que cuando la extensión del gerente de la compañía timbre y este no conteste la llamada se redirija a su celular y otros teléfonos de contacto. Se pueden especificar tiempos de gracia antes de que el siguiente número de teléfono sea marcado.

4.4 Funcionamiento de Asterisk

Asterisk organiza sus archivos distribuyéndolos entre algunos directorios. Los más importantes son los siguientes.

Directorio	Descripción
<code>/etc/asterisk/</code>	Lugar donde residen los archivos de configuración de Asterisk
<code>/usr/lib/asterisk/modules/</code>	Contiene los módulos de Asterisk
<code>/usr/sbin/</code>	Incluye el binario de Asterisk
<code>/var/lib/asterisk/agi-bin/</code>	Lugar donde residen los <i>scripts</i> AGI
<code>/var/lib/asterisk/mohmp3</code>	Contiene archivos para música en espera
<code>/var/lib/asterisk/sounds</code>	Incluye sonidos que Asterisk utiliza como <i>prompts</i> de voz
<code>/var/spool/asterisk/</code>	Lugar donde Asterisk guarda archivos que genera producto de su funcionamiento como <i>voicemails</i> y grabaciones de llamadas
<code>/var/run/</code>	Guarda archivos con información de procesos (PIDs)
<code>/var/log/asterisk/</code>	Aquí residen los archivos de <i>log</i> de Asterisk como los CDRs o el <code>/var/log/asterisk/full</code>

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 si solo no cumpliría ninguna función interesante.

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

Los módulos pueden ser controlados a través del archivo de configuración `/etc/asterisk/modules.conf`. Este archivo instruye a Asterisk a que cargue o no un módulo específico a través de las directivas `load` y `noload`. Por omisión en Elastix todos los módulos tratan de ser cargados.

A continuación un fragmento del archivo `modules.conf` que viene por defecto con Elastix. Aquí se puede observar cómo se le instruye a Asterisk a 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

; 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 (no confundir con la línea de comandos de Linux) con el comando:

```
module load <modulo>
```

Más adelante veremos más de la consola de Asterisk o CLI.

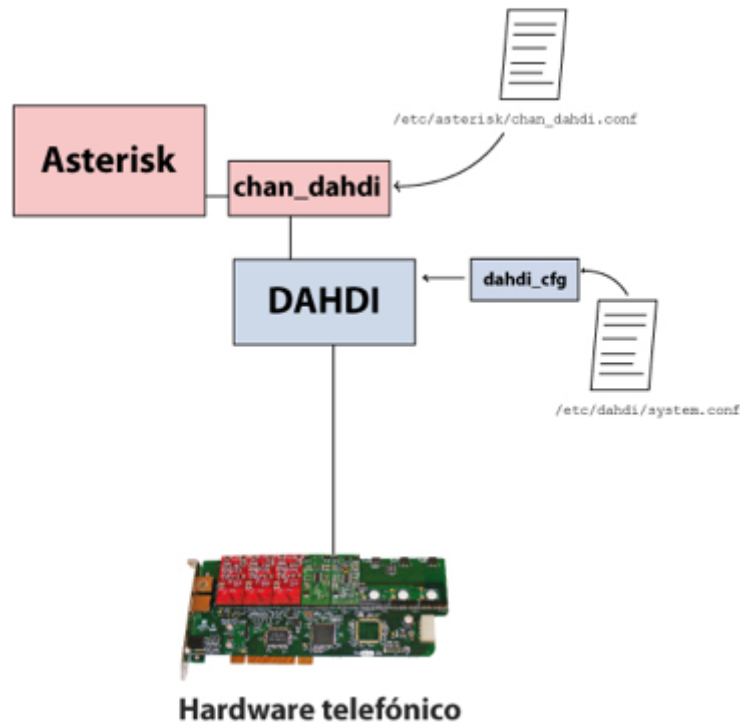
4.5 El proyecto DAHDI

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

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

Estos drivers se comunican con Asterisk a través de un módulo de Asterisk llamado `chan_dahdi.so` que se configura a través del archivo de configuración `/etc/asterisk/chan_dahdi.conf`.

Adicionalmente DAHDI tiene sus propios archivos de configuración, independientes de Asterisk, que se encuentran en la carpeta `/etc/dahdi/`.



- Nota: En versiones de Elastix menores a 1.5 se utiliza el software del proyecto Zaptel en lugar de DAHDI. Esto quiere decir que los archivos de configuración para los drivers de telefonía tienen nombres diferentes.

Breve historia de DAHDI

A pesar del cambio de nombre de Zaptel a DAHDI y toda la confusión que esto generó en un principio, no hay que olvidar que el origen de DAHDI estuvo en el proyecto Zaptel. Primero porque hay que dar el crédito merecido a quienes trabajaron en el proyecto original y segundo para no olvidarnos de la historia. A continuación un breve resumen de dicha historia.

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 Telephony. No paso mucho tiempo hasta que Digium comenzó a mejorar los drivers y extender el soporte para nuevos modelos de hardware, entre ellos los fabricados por Digium mismo. Es así que 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 por una compañía vendedora de tarjetas de llamadas y para evitar cualquier posible confusión futura decidió cambiar de nombre a sus drivers. El nombre elegido es DAHDI, que es un acrónimo de Digium Asterisk Hardware Device Interface o Interfaz de Dispositivos de Hardware de Asterisk de Digium en español.

Digium finalizó el soporte oficial de Zaptel a partir de la versión Asterisk 1.6

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. Debido a esto, podemos usar el editor vim u otro editor de línea de comandos de nuestra preferencia para editarlos.

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

Archivo	Descripción
<code>extensions.conf</code>	Define el <i>plan de marcado</i> . En Elastix se incluyen otros archivos más para organizar el plan de marcado de mejor manera. Estos archivos adicionales empiezan con la cadena <i>extensions_</i>
<code>sip.conf</code>	Configura los dispositivos SIP
<code>iax.conf</code>	Configura los dispositivos IAX
<code>chan_dahdi.conf</code>	Define la configuración de los canales tipo DAHDI.

Comentarios en los archivos de configuración

Los archivos de configuración de Asterisk pueden contener comentarios. Un comentario empieza con el caracter de punto y coma. Una vez escrito este caracter el resto de la línea se convierte en un comentario. Es decir, esta línea es ignorada 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 se incluyen otros importantes archivos que contienen también porciones del plan de marcado. Los principales son:

- **`extensions_additional.conf`**: Aquí reside el plan de marcado variable escrito por FreePBX, el cual es un componente importante de Elastix. Si se realizan cambios manualmente en este archivo, los mismos se perderán cuando FreePBX lo sobrescriba.
- **`extensions_custom.conf`**: Aquí el usuario puede definir manualmente el plan de marcado. Este archivo no es tocado por FreePBX por lo que es seguro editarlo sin temor a perder los cambios.

Todos estos archivos de plan de marcado están escritos en texto plano en un formato definido por Asterisk. El contenido de estos archivos se encuentra dividido en secciones llamadas contextos.

Contextos

Los contextos sirven para agrupar la lógica del plan de marcado. Los contextos también a su vez incluyen otros contextos mediante la cláusula “include” o redirigir el hilo de ejecución del plan de marcado a otros contextos dependiendo de condiciones. Por ejemplo, luego de examinar el prefijo del número telefónico marcado, un contexto puede contener la lógica necesaria para rutear una llamada hacia la troncal correspondiente. 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 se encuentra 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] .

Extensiones

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

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

- *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 funciones mucho más complejas.*

Una extensión típica 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 el asunto.

Existe una extensión especial no numérica llamada extensión **s**, a la 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 realizó la llamada aun no ha presionado el número de la extensión. Por ejemplo, este es el caso de 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 extensión 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, revisemos 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")
```

Este plan de marcado 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 funciones.

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, 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, de canal y pre-definidas.

Globales

Son las variables definidas en la sección [globals] del archivo /etc/asterisk/extensions.conf, o directamente dentro de /etc/asterisk/globals_custom.conf en cualquier Elastix. 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 pre-definidos y contienen cierta información dependiendo del comportamiento de la llamada o de otros factores. Algunas variables pre-definidas son:

```
${CALLERID(all)}
```

```
${CALLERID(name)}
```

```
${CALLERID(num)}
```

```
${CHANNEL}
```

```
${CONTEXT}
```

```
${EPOCH}
```

```
${EXTEN}
```

```
${SIPUSERAGENT}
```

```
${UNIQUEID}
```

- Nota: Hay un tipo adicional de variable introducida a partir de Asterisk 1.6 llamada variable compartida o *shared*. Esta variable permite que dos o más canales accedan a lo que de otra manera sería una variable de canal.

A continuación reproducimos un segmento del archivo `extensions.conf` que se distribuye por defecto con Elastix. Aquí podemos ver 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 inmediatamente si éste está timbrando. Puede recibir opcionalmente el parámetro delay que indica a Asterisk que debe esperar un número determinado de milisegundos después de contestar la llamada para pasar a la siguiente aplicación.

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 es reproducido. Cuando termina de reproducir el último archivo, su ejecución finaliza por lo que si se quiere seguir esperando por una extensión se debe usar la aplicación WaitExten() en conjunto.. 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 el valor -1.

Goto

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

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

Dial

Sintaxis: Dial(type/identifíer, timeout, options, URL)

Este comando es el responsable de marcar 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. Además, 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 nivel de debug es 0 y el de verbose 3 en Elastix.

- Nota: Debido a la gran cantidad 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 del sistema. En sistemas con mucha actividad este consumo adicional puede ser significativo y afectar el desempeño de las llamadas.

Comandos CLI

A continuación proveemos una breve lista de los comandos de CLI más comunes.

Comando	Descripción
<code>agi set debug on</code>	Permite habilitar el <i>debug</i> de <i>scripts</i> AGI. Se apaga con “ <code>agi set 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 actualizada de los canales activos
<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 IAX. Se deshabilita con el comando “ <code>iax2 set debug off</code> ”
<code>iax2 show peers</code>	Muestra los <i>peers</i> definidos y alguna información adicional como la dirección IP desde donde se conectan, el estatus, entre otros datos
<code>iax2 show registry</code>	Muestra información de registro IAX e información como el estatus y la dirección 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 set debug on</code>	Habilita <i>debug</i> de los spans de la interfaz PRI. Se apaga con “ <code>pri set debug off</code> ”

<code>core restart gracefully</code>	Reinicia Asterisk de manera ordenada. Es decir, no permite iniciar nuevas llamadas y espera a que las existentes terminen para proceder a reiniciar
<code>core restart now</code>	Reinicia Asterisk de manera brusca (de inmediato)
<code>core 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 que se corten llamadas en curso
<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 <i>debug</i> 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 su estatus
<code>sip show registry</code>	Muestra un listado con los registros SIP e información relevante de cada uno de ellos
<code>core stop gracefully</code>	Detiene Asterisk de manera ordenada. Es decir, no permite establecer nuevas llamadas y espera a que las existentes terminen para proceder a detener el Asterisk.
<code>core stop now</code>	Detiene Asterisk de manera brusca (de inmediato)
<code>core stop when convenient</code>	Detiene Asterisk cuando no haya actividad en la central. Adecuado en ambientes de producción donde no tenemos otra opción que apagar la central pero no queremos que se corten llamadas en curso
<code>dahdi show status</code>	Muestra un listado de los canales DAHDI y un reporte de alarmas y errores
<code>dahdi show channels</code>	Muestra un listado de los canales DAHDI y algunos parámetros de funcionamiento.

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

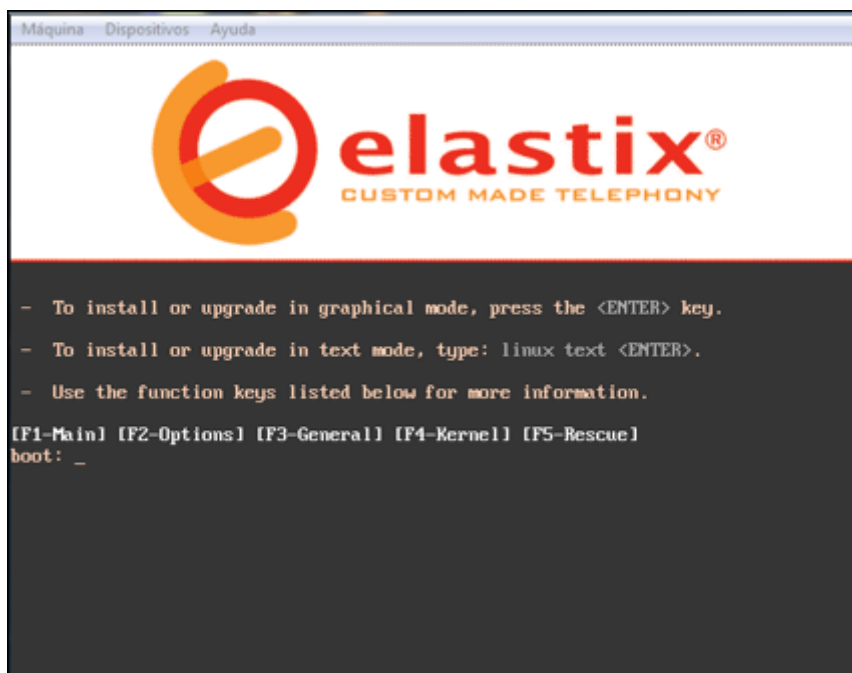
5.1 Instalando desde CD

Antes de comenzar, asegúrese de tener una copia de la última versión estable de Elastix, (la presente obra está basada en la versión 2.0). Ésta puede ser descargada desde la página de descargas del proyecto Elastix en www.elastix.org. Escoja la versión apropiada dependiendo si dispone de una máquina con procesador de 32 o 64 bits.

Arranque desde CD

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

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

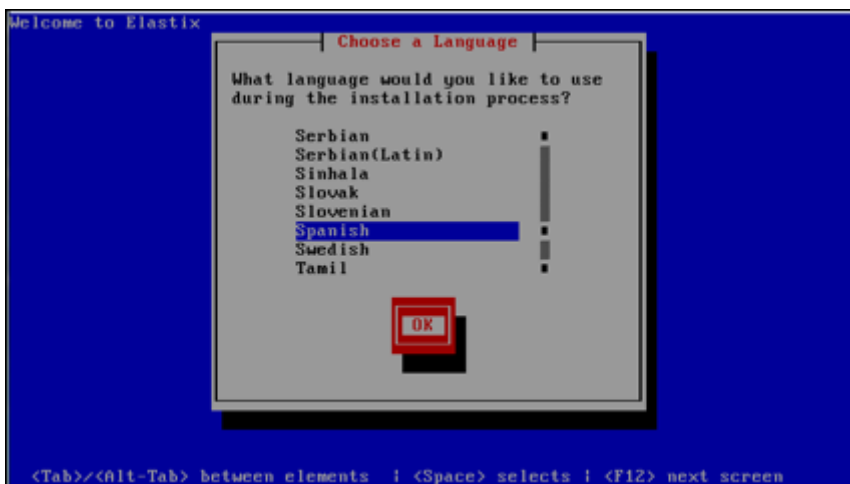


- Nota: El CD de instalación de Elastix formateará TODO el disco duro durante el proceso de instalación así que asegúrese de no tener información que vaya a necesitar en su disco duro.

El proceso de instalación iniciará automáticamente dentro de diez segundos. Usted podría presionar la tecla de Enter si no desea esperar este tiempo.

Preferencias y bienvenida

Luego de cargar algunos programas necesarios para iniciar la instalación, el proceso nos solicitará el idioma que será utilizado para formular preguntas y mostrar mensajes a lo largo del proceso de instalación. Escoja el idioma español (Spanish) utilizando las teclas de flechas de su teclado. Usando la tecla de TAB posicione el cursor sobre OK y presione la tecla de Enter.



A continuación, proceda a escoger el tipo de teclado de acuerdo al idioma. Si su teclado es de idioma español seleccione la opción es.



Particiones del disco duro

El siguiente es el paso crítico de la instalación ya que configura las particiones del disco duro. Si usted es un administrador experimentado, podría escoger la opción Utilizar diseño personalizado. Caso contrario, elija Usar espacio disponible en dispositivos seleccionados y crear diseño predeterminado. En nuestra instalación seleccionaremos esta última opción.

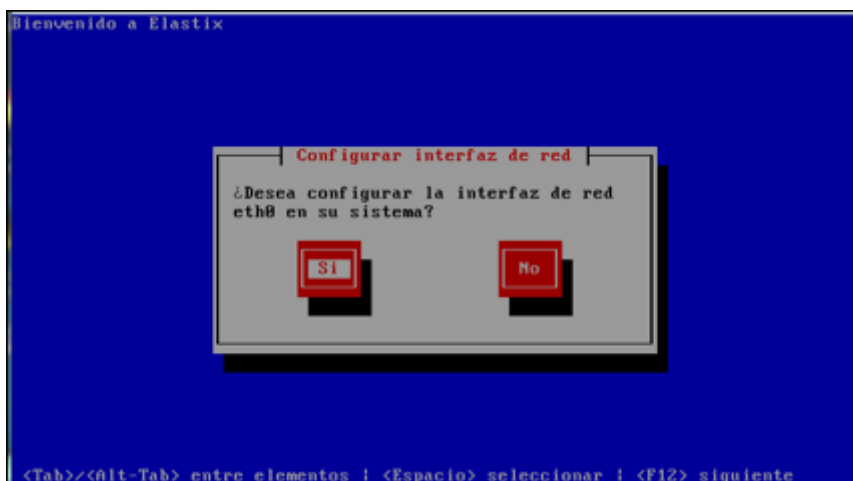


Ahora el programa instalador nos consulta si queremos revisar las particiones creadas. Escoja la opción No y digite la tecla de Enter para proseguir.



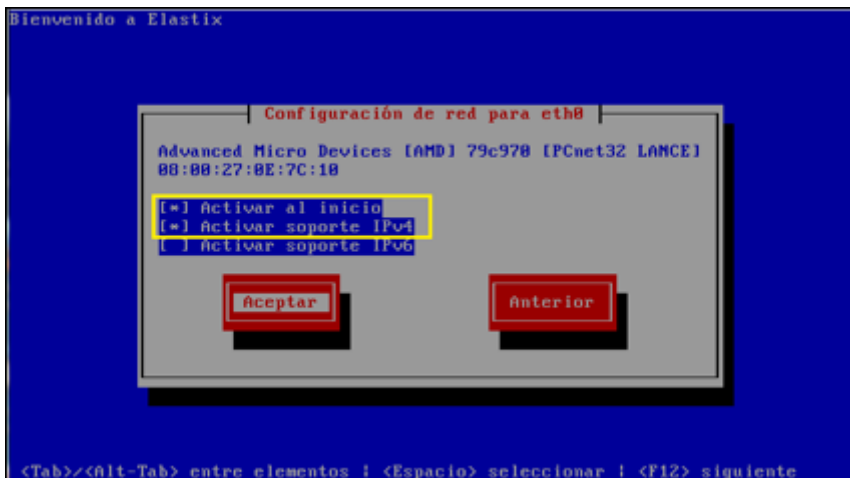
Configuración de parámetros de red

Es el turno de la configuración de los parámetros de red. El programa de instalación preguntará si desea configurar los parámetros de direccionamiento IP en las tarjetas de red con las que cuenta su servidor. En la siguiente pregunta escoja Si.

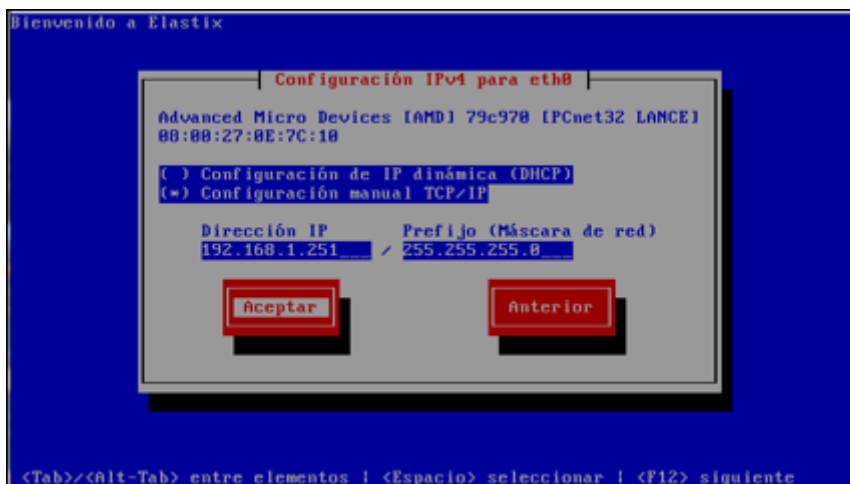


Para cada tarjeta de red que se tenga instalada se preguntará si la desea activar a tiempo de

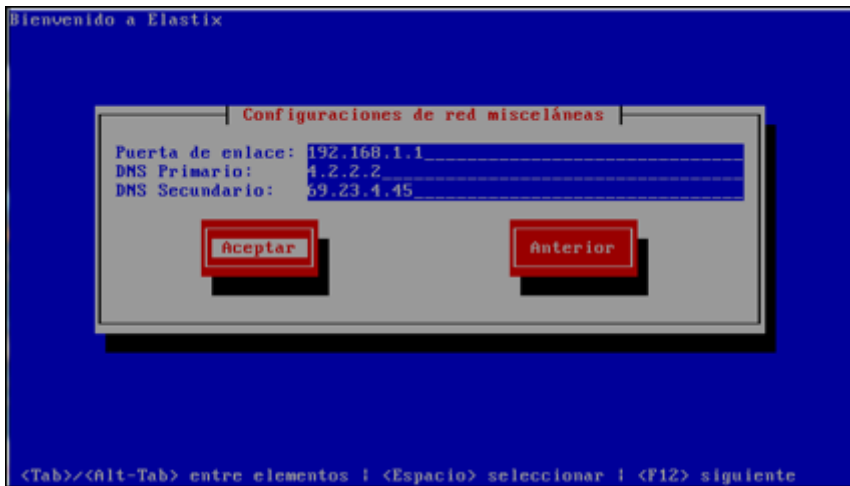
arranque y qué tipo de soporte IP se va a habilitar en ella. El protocolo IP que hemos abordado en el capítulo 2 es la versión 4, por lo que recomendamos escoger el soporte de IPv4. Presione Aceptar.



Si usted cuenta con un servidor de Dynamic Host Configuration Protocol (DHCP) en su red para la asignación dinámica de direcciones IP, seleccione la opción Configuración de IP dinámica. De lo contrario deberá asignar la dirección IP de manera manual seleccionando la opción Configuración manual TCP/IP.



La siguiente pantalla nos pide el ingreso de parámetros de red adicionales como la puerta de enlace (gateway), DNS primario y DNS secundario. Continúe presionando Aceptar.



A continuación debemos bautizar nuestro servidor con un nombre de host o hostname. Se debe considerar que un nombre host correcto tiene la forma nombre.dominio. Su administrador de red sabrá decirle que nombre de host usar. En caso de no contar con uno, ingrese un nombre ficticio ya que luego es posible cambiarlo. Escoja Aceptar para continuar.



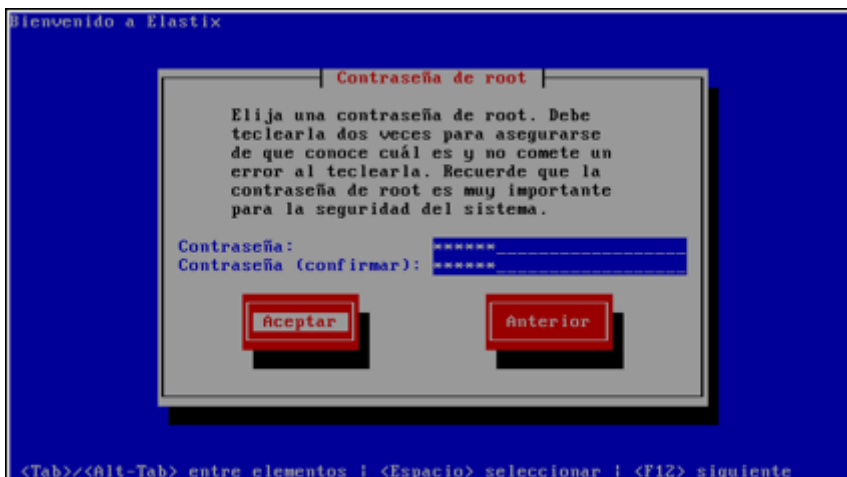
Huso horario, contraseña de root y paquetes

En la siguiente pantalla, seleccione el huso horario de su región y presione Aceptar para continuar.



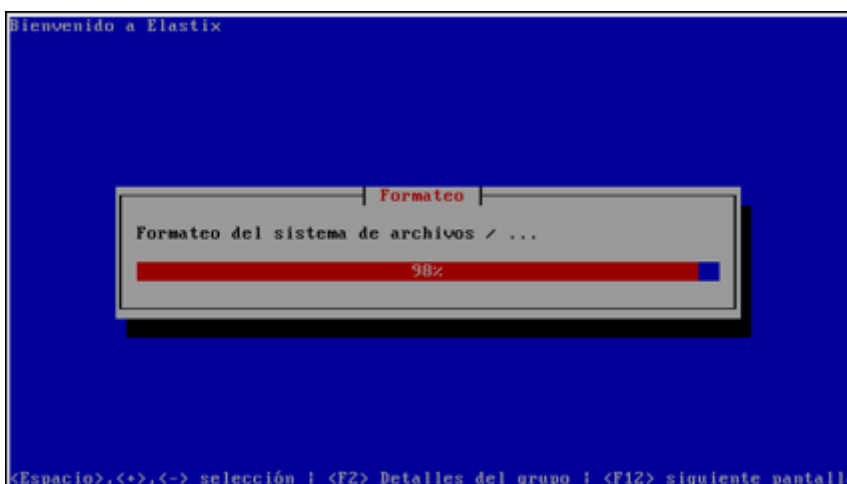
Es hora de especificar la contraseña que será usada por el usuario con privilegios de administrador de Elastix. Recuerde esta contraseña ya que es una parte crítica para la seguridad del sistema. Debido a

la importancia de este paso, se requiere confirmar la contraseña ingresándola nuevamente. Presione Aceptar para seguir con la instalación.

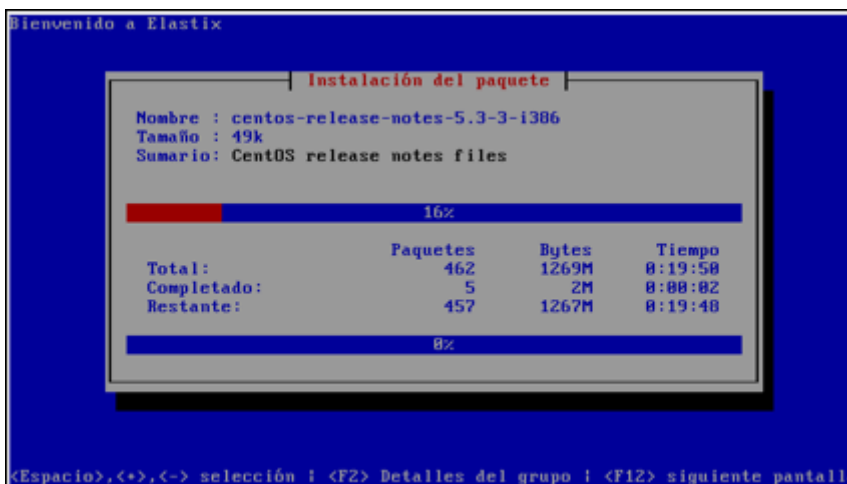


Formateo y copia de archivos.

A partir de este momento el programa instalador particionará y formateará el disco duro del servidor y toda la información almacenada allí se perderá para siempre.



Adicionalmente, se copiarán todos los archivos necesarios para correr Elastix hacia el disco duro.



Una vez que termina la copia de paquetes al disco duro, el servidor se reiniciará automáticamente.

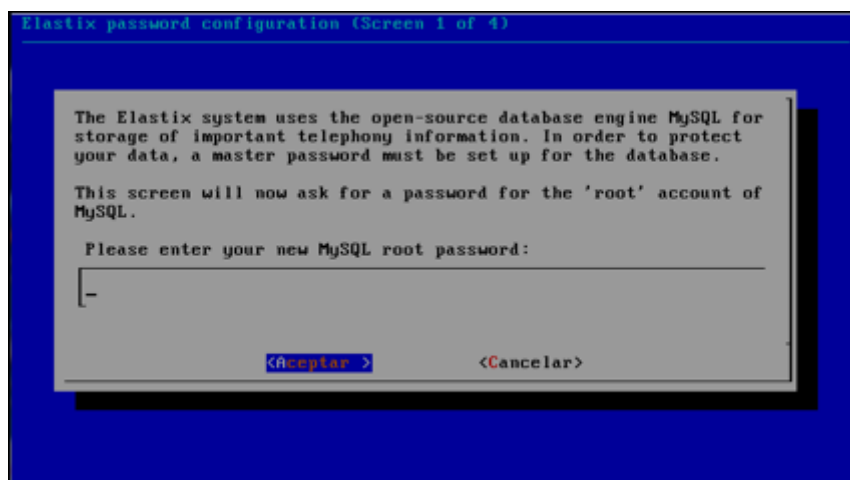
Reinicio automático y contraseñas varias

Permita que el sistema arranque por sí solo asegurándose de no presionar ninguna tecla en la siguiente pantalla.

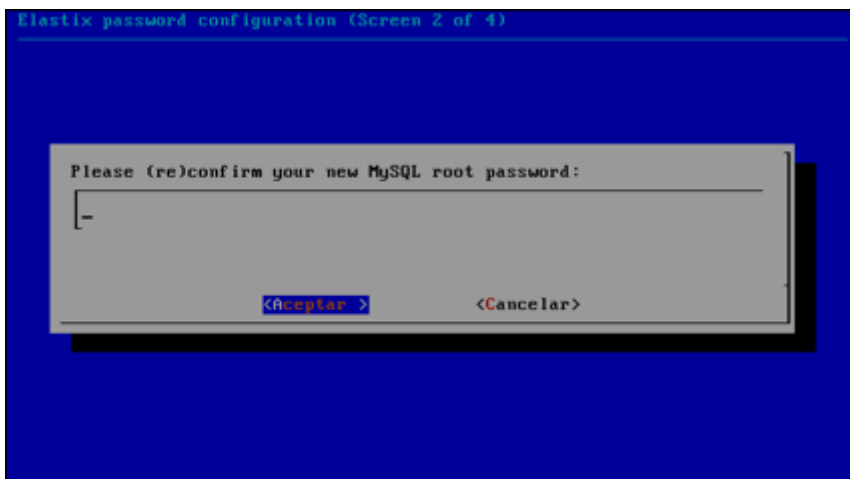


Luego de cargar programas esenciales para su funcionamiento, Elastix procederá a solicitar contraseñas para los componentes importantes que conforman su sistema. Esta fase de ingreso de claves solo se realizará en el primer arranque luego de la instalación.

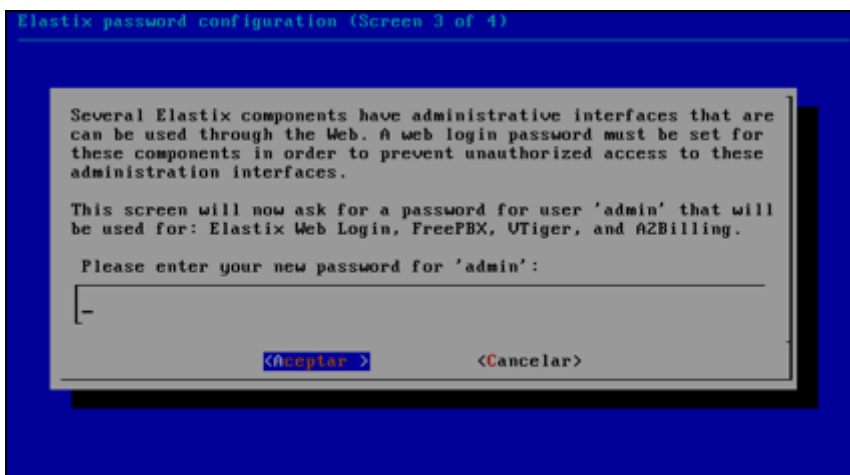
La primera clave que se solicita es la del administrador para la base de datos MySQL. Ingrese una clave y continúe con el botón de Aceptar



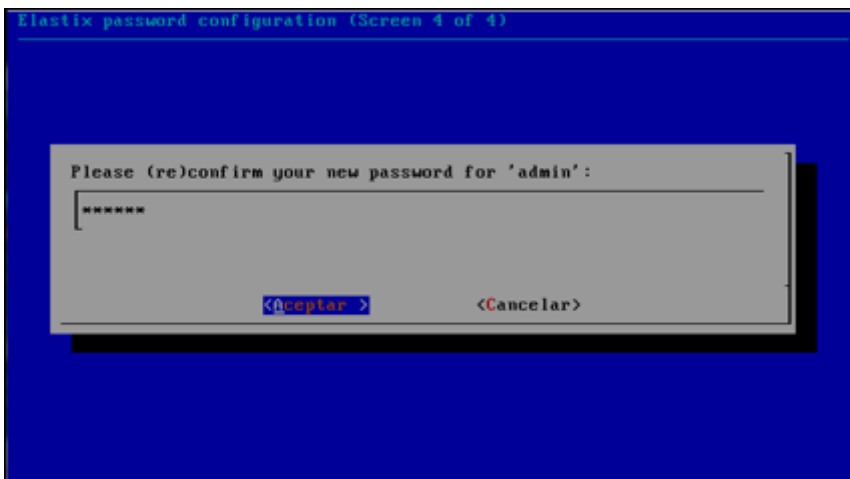
Usted deberá confirmar la clave anteriormente ingresada para administración de MySQL nuevamente en la siguiente pantalla.



La siguiente contraseña será utilizada para ingresar con privilegios de administrador a la interfaz gráfica de Elastix. Además, esta misma clave será utilizada para ingresar a los componentes FreePBX, VTiger y A2Billing del sistema. Una vez digitada la clave, avance a la siguiente pantalla seleccionando Aceptar.



Al igual que en el caso anterior, usted deberá confirmar la clave anteriormente ingresada para Elastix y sus componentes. Luego de ingresarla nuevamente, presione Aceptar para continuar.



Finalizar la instalación

El proceso de instalación culminará cuando usted reciba la pantalla de invitación a ingresar a la consola principal del servidor, tal como se muestra en la figura siguiente.

```
CentOS release 5.5 (Final)
Kernel 2.6.18-194.3.1.el5 on an i686

pbx login: _
```

Para acceder a la consola de Elastix, simplemente ingrese root en el prompt de pbx login. La contraseña es la que usted eligió durante el proceso de instalación.

6 Introducción a Elastix

“Los que dicen “imposible” no deberían interrumpir a los que están intentando”

- Proverbio Chino

La intención del presente capítulo es la de hacer un sobrevuelo rápido por Elastix, tratando de aclarar los conceptos básicos antes de abordar temas más avanzados. Como el lector podrá notar no se trata de un capítulo muy técnico sino más bien informativo a nivel general.

6.1 ¿Qué es Elastix?

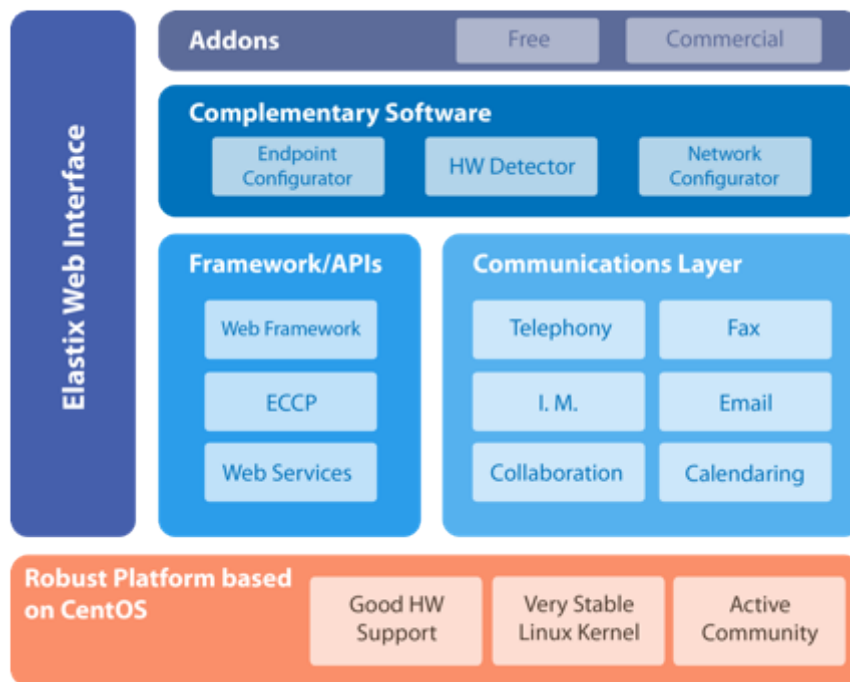
Elastix es una distribución de software libre de Servidor de Comunicaciones Unificadas que integra diferentes tecnologías de comunicaciones en un solo paquete como:

- PBX
- Fax
- Email
- Mensajería Instantánea (IM)
- Calendario
- 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 el sistema operativo. Una vez instalada la distro tendremos un Servidor de Comunicaciones Unificadas listo para producción.

Elastix implementa gran parte de su funcionalidad sobre cuatro programas de software muy importantes como son Asterisk, Hylafax, Postfix y Openfire. Estos programas brindan las funciones de PBX, Fax, Email y Mensajería Instantánea, respectivamente. El sistema operativo se basa en la popular distribución de Linux orientada a servidores llamada CentOS.

A continuación se muestra una figura donde se pueden observar los componentes de Elastix y su relación entre sí.



6.2 Breve historia del proyecto

Elastix fue creado y actualmente es mantenido por la compañía ecuatoriana PaloSanto Solutions. Elastix fue liberado como software libre por primera vez en Marzo de 2006. Inicialmente no se trataba de una distro sino más bien de una interfaz para mostrar registros de detalles de llamadas para Asterisk (CDRs). Fue recién a finales de Diciembre de 2006 cuando se lo lanzó como una distribución que contenía diversas herramientas administrables bajo una misma interfaz Web y que llamó la atención por su usabilidad.

Desde entonces hasta la fecha Elastix no ha parado de crecer en popularidad, habiendo superado en 2010 su primer millón de descargas.

Galardones

En 2007 el proyecto estuvo nominado en 2 categorías para los premios Community Choice Awards (CCA) de SourceForge y volvió a ser nominado en 2008 y 2009. En Febrero de 2011 recibió el premio a Mejor Producto del año 2010 otorgado por la compañía de medios TMC, quienes además publican la revista Internet Telephony.

6.3 Características de Elastix

Es difícil enlistar todas las características de Elastix debido a que constantemente se añaden más funcionalidades. Sin embargo, las más sobresalientes basados en Elastix 2.0 son las siguientes:

PBX

1. Voicemails vía Web o email
2. Grabación de llamadas
3. IVR completo y altamente flexible
4. Parqueo de llamadas
5. Colas de atención
6. Soporte de síntesis de voz
7. Identificación de llamadas
8. Horarios de atención
9. Provisión automática de teléfonos vía Web
10. Creación de lotes de extensiones
11. Soporte para video vía H.263 y H.264
12. Detección automática de hardware de telefonía
13. Servidor DHCP embebido
14. Reporte de detalle de llamadas (CDRs)
15. Tarifación con reportación de consumo por destino
16. Reporte de uso de canales por tecnología (SIP, ZAP, IAX, Local, H323)
17. Soporta protocolos SIP, IAX, H323, MGCP, iLBC, entre otros
18. Soporta codecs ADPCM, G.711 (A-Law & μ -Law), G.722, G.723.1 (pass through), G.726, G.729, GSM, iLBC
19. Soporte para interfaces análogas FXS/FXO
20. Soporte para interfaces digitales E1/T1/J1 (PRI/BRI/R2)
21. Soporte para grupos de troncales
22. Follow-me, grupos de ringado, paging, DISA, Call back, PINes de seguridad, entre otros

Fax

1. Servidor de Fax administrable desde Web
2. Visor de Faxes integrado con descargas en PDF
3. Integración fax-a-email
4. Envío de fax vía Web
5. Control de acceso para clientes de fax
6. Capacidad de integración con WinprintHylafax para permitir imprimir-a-fax desde Windows.

Documentación

- [Servicio de FAX en Elastix](#)
- PDF [Enviar y recibir fax en Elastix con cliente Windows](#)

Email

1. Servidor de Email multi-dominio administrable vía Web
2. Soporte para SMTP saliente
3. Configuración de email relay
4. Cliente de Email basado en Web
5. Soporte para cuotas en las cuentas de email
6. Funcionalidad de listas de distribución vía Mailman

Documentación

- [Tutorial Elastix Español - Servidor de correo](#)
- [Configurar Elastix para enviar correo desde una cuenta en otro servidor](#)

Mensajería instantánea (IM)

1. Servidor de mensajería instantánea basado en OpenFire integrado
2. Llamada vía PBX a usuarios conectados
3. Configuración completamente basada en Web
4. Interconexión simultánea a Yahoo, MSN Messenger, GTalk, ICQ, entre otros
5. Soporta conexiones server-to-server para compartir usuarios
6. Reporte de sesiones de usuarios
7. Soporte para plugins
8. Soporte para LDAP

Documentación

- PDF [guia-de-configuracion-de-modulo-sms-en-elastix.pdf](#)
- Video [Mensajería Instantánea con Elastix](#)

Calendario

1. Interfaz de calendario vía Web
2. Recordatorios de voz
3. Envío de invitaciones a eventos vía email
4. Exportación de calendario a formato iCalendar

Colaboración

1. Libreta telefónica (Phone Book) con capacidad para click-to-call
2. Funcionalidad de CRM integrada vía vTigerCRM
3. Soporte para teleconferencias de audio configurable vía Web
4. Funcionalidad de conferencias Web vía Addons

Call Center

1. Soporte para agentes con logon/logoff
2. Marcación predictiva
3. Formularios genéricos para ingreso de datos
4. Disponibilidad de reportes detallados por agente

General

1. Ayuda en línea embebida
2. Interfaz Web con soporte multi-lenguaje
3. Visualización del sistema vía Dashboard
4. Configuración de fecha/hora/uso horario
5. Configuración de parámetros de red
6. Configuración de usuarios y permisos
7. Configuración de respaldos
8. Configuración de temas o skins
9. Actualización centralizada de software
10. Apagado y encendido remoto vía Web
11. Visualización del sistema en tiempo real vía Operator Panel

6.4 Licenciamiento

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

Para acceder al texto completo de la licencia el lector puede seguir el siguiente [vínculo](#).

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>

En este sitio el usuario puede encontrar herramientas de soporte útiles como:

- [Foros](#)
- [Wiki](#)
- [Links a descargas](#)

Foros

Una de las herramientas más importantes en el sitio Web Elastix.org 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 desorganizar el foro y el nuevo comentario podría ser ignorado.
- Use un lenguaje adecuado para no herir la susceptibilidad de nadie. Si su mensaje contiene lenguaje subido de tono, amenazante o mal intencionado, su mensaje puede ser eliminado permanentemente. La amabilidad es un buen ingrediente si desea que otras personas colaboren con su problema.
- El foro no es un espacio publicitario.
- 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 vínculo.
- 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.

6.6 La interfaz Web de Elastix

Finalmente llegó el momento de ingresar a la interfaz Web de Elastix. Bueno, realmente no creo que el lector haya llegado hasta aquí sin haber ingresado antes a la interfaz de administración Web (por lo menos para curiosidad), pero quería decirlo de esta manera para expresar la importancia de este momento.

La interfaz Web de administración es el componente que define a Elastix. Sin este componente Elastix no fuera lo que es, ya que uno de los objetivos del producto es la facilidad de uso. A menos que se trate de alguna configuración personalizada desde aquí es posible administrar prácticamente todo.

Se puede utilizar cualquier navegador de Internet para acceder a esta interfaz. Sin embargo, dado que el equipo de desarrollo de Elastix usa principalmente Firefox para sus pruebas, recomendamos el uso de este navegador.

Para ingresar debemos apuntar a la dirección IP del servidor Elastix desde el navegador y veremos una pantalla como la siguiente.

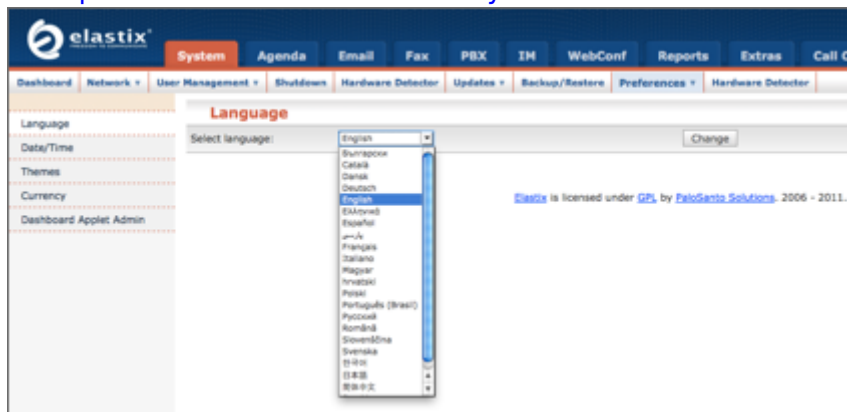


Al momento de instalación el sistema generó de manera automática un usuario llamado "admin". Este usuario tiene privilegios de administrador y nos servirá para ingresar por primera vez. Notar que el nombre de usuario se encuentra todo en minúsculas. La contraseña para ingresar es la misma que se ingresó durante el proceso de instalación (ver capítulo 5).

Una vez validado el nombre de usuario y contraseña el sistema nos mostrará la interfaz de administración. Lo primero que veremos será el Dashboard que es básicamente una pantalla de resumen con algo de estadística de uso de recursos y estado actual de los procesos más importantes del sistema.



Lo siguiente es personalizar un poco la interfaz a nuestras necesidades. Algunas sugerencias son: asegurarnos que el idioma a utilizar sea el Español, que la fecha y hora del sistema, así como el huso horario, se encuentren correctos, e inclusive podríamos jugar un poco cambiando el look-and-feel de la interfaz. Todo esto se puede hacer desde el menú "System → Preferences".



6.7 Configuración de red

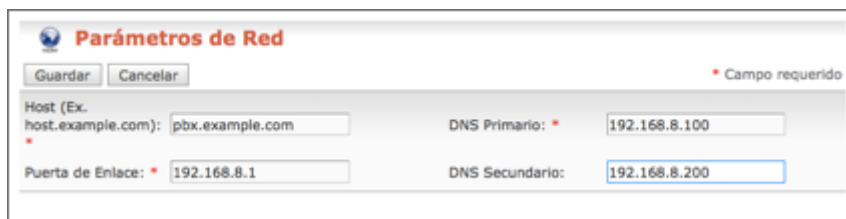
Usualmente la primera cosa importante que se debe configurar luego de ingresar al equipo por primera vez son los parámetros de red debido a que por omisión Elastix usa un servidor DHCP para obtener una dirección IP y esto no es lo ideal si vamos a montar un servidor de comunicaciones unificadas. Lo deseable es usar una dirección IP estática.

La configuración de los parámetros de red puede hacerse desde el menú "Sistema → Red". Aquí podremos configurar dos cosas: los parámetros generales de red y los parámetros asociados con cada una de las tarjetas de red que el servidor tenga instaladas, incluyendo las que vienen embebidas en la placa madre o mainboard.



Para editar los parámetros generales de red damos clic en el botón “Editar Parámetros de Red” que aparece en la parte superior de la pantalla anterior.

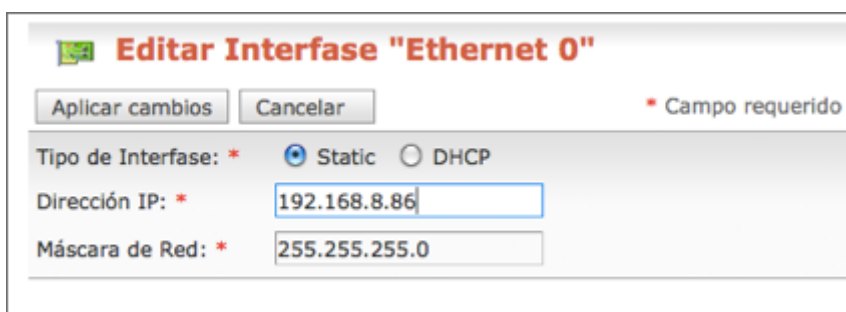
Esto nos conducirá a otra pantalla que se muestra a continuación.



A continuación se explica el significado de cada uno de los campos solicitados.

- 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.

Para editar los parámetros asociados con cada tarjeta de red debemos dar clic el nombre de dicha interfaz. En nuestro ejemplo daremos clic donde dice “Ethernet 0” y seremos conducidos a la siguiente pantalla.



Como podemos observar aquí podemos cambiar la asignación de IP de DHCP a Estática que es lo que queríamos.

6.8 Sobrevuelo por la interfaz de administración Web de Elastix

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

En esta sección haremos un sobrevuelo de la funcionalidad más importante que podemos administrar desde esta interfaz. Además explicaremos cómo se encuentran organizados los menús y módulos con

los que cuenta Elastix.

Menús y módulos

La interfaz Web de Elastix se encuentra organizada en menús. Así tenemos los menús de primer nivel como: Sistema, Agenda, Email, Fax, PBX, IM, etc., y también tenemos los menús de segundo nivel, también llamados sub-menús.

Un menú se puede entender como un contenedor que sirve para organizar la interfaz de Elastix.

Un módulo es cambio es un componente de Elastix que nos ofrece algún tipo de funcionalidad. Los módulos son contenidos por menús o sub-menús para de esta forma ser organizados dentro de la interfaz Web y que el usuario pueda acceder a ellos navegando entre los menús.

En el siguiente ejemplo se puede observar que el módulo “Fecha/Hora” se encuentra contenido dentro del menú (o sub-menú) “Preferencias”, el cual a su vez se encuentra contenido dentro del menú “Sistema”. La ubicación del módulo puede entonces ser referida como “Sistema à Preferencias à Fecha/Hora”.



En ocasiones un módulo puede estar contenido dentro de solo un menú como en la siguiente figura.



- Nota: Es importante tener presente el concepto de módulo y menú, especialmente si se está interesado en modificar el código de Elastix para realizar alguna personalización.

Navegando por la interfaz Web

A continuación navegaremos por la interfaz Web de Elastix, explicando brevemente el significado de cada menú y módulo. Cada menú de primer nivel aparecerá encabezando la tabla y cada submenú

aparecerá en negritas.

- *Nota: Es importante mencionar que en el siguiente listado sólo se incluyen los menús que se instalan por defecto. Es decir que no se incluyen menús que se instalan a través de Addons como el módulo de Call Center o el de Web Conference.*

System

Dashboard	<i>Dashboard</i> es una especie de escritorio donde el usuario puede ver un resumen de la actividad en Elastix como sus últimos <i>emails</i> , <i>voicemails</i> , faxes, y citas de calendario. El usuario administrador tiene una interfaz diferente donde además puede monitorear el rendimiento del sistema y otras funciones útiles con solo añadir <i>applets</i> .
Network	Configuración de parámetros de red
à Network Parameters	Especificar dirección IP y máscara de red, <i>gateway</i> , nombre de <i>host</i> , servidores DNS, entre otros.
à DHCP Server	Habilitar el servidor DHCP embebido que viene con Elastix
à DHCP Client List	Mostrar lista de clientes DHCP conectados
à Assign IP Address to Host	Asignar dirección IP a un cliente específico
User Management	Administración de usuarios de Elastix
à Groups	Especificar grupos de usuarios
à Users	Administrar usuarios y asignarlos a grupos. También permite asociar cuentas de <i>email</i> y extensiones telefónicas a los usuarios
à Group Permissions	Configuración de los permisos de acceso a los diferentes módulos para un grupo determinado
Shutdown	Apagar o reiniciar el servidor
Hardware Detector	Detección de hardware telefónico
Updates	Actualización de paquetes de software
à Packages	Listado de paquetes con la opción de instalar o actualizar
à Repositories	Configuración de los repositorios en base a los cuales se realizan las actualizaciones
Backup/Restore	Generar respaldos y restituir información para el servidor Elastix
Preferences	Especificar las varias preferencias del sistema
à Language	Cambio del idioma de toda la interfaz Web de Elastix

à Date/Time	Cambio de la fecha, hora y zona horaria del servidor
à Themes	Cambio de los temas (<i>skins</i>) para darle a la interfaz de Elastix un <i>look</i> diferente
à Currency	Cambio del símbolo de la divisa que se muestra en los reportes de tarificación
à Dashboard Applet Admin	Selección de los <i>applets</i> incluidos en el Dashboard del sistema

Agenda

Calendar	Agendar eventos en formato calendario que inclusive pueden generar llamadas telefónicas automáticas
Address Book	Libreta de direcciones para contactos
Recordings	Grabación de mensajes que se pueden asociar con eventos agendados en el módulo Calendar

Email

Domains	Creación de dominios de <i>email</i> . Elastix soporta múltiples dominios
Accounts	Administración de cuentas de email y asignación de cuotas de espacio en disco duro
Relay	Configuración de <i>relay</i> permite usar Elastix para enviar email desde otras redes
Webmail	Cliente de email basado en Web vía Roundcube
Antispam	Habilitar detección de correo SPAM y especificar la sensibilidad
Remote SMTP	Configurar salida de emails por medio de un servidor SMTP remoto
Email list	Añadir soporte para listas de distribución (<i>mailing lists</i>) vía Mailman

Fax

Virtual Fax	Administración de faxes virtuales
à Virtual Fax List	Listado de extensiones de fax virtuales que recibirán faxes en formato PDF vía <i>email</i>
à New Virtual Fax	Creación de extensiones de fax nuevas
à Send Fax	Envío de fax desde la interfaz misma como texto simple o documento adjunto
Fax Master	Establecimiento de 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 Viewer	Visualización de faxes en formato PDF
Email Template	Configuración de plantilla de <i>email</i> que se enviará cada vez que arribe un fax

PBX

PBX Configuration	Aplicación FreePBX embebida permite la mayoría de configuraciones a nivel de central telefónica
Operator Panel	Permite la visualización de extensiones, troncales y colas en uso en tiempo real
Voicemail	Listado de <i>voicemails</i> . Se debe haber asociado previamente al usuario con una extensión telefónica para poder verlo
Monitoring	Listado de grabaciones telefónicas. Se debe haber asociado previamente al usuario con una extensión telefónica para poder verlo
Endpoint Configurator	Permite provisionar un gran número de teléfonos en corto tiempo
Conference	Generar teleconferencias de audio
Batch of Extensions	Permite crear un gran número de extensiones en corto tiempo
Tools	Acceso a varias herramientas útiles
à Asterisk-Cli	Ejecución de comandos del CLI
à Asterisk File Editor	Edición de archivos de texto
à Text to Wav	Conversión de texto a audio
Flash Operator Panel	Alternativa de panel de operador basado en flash.
VoIP Provider	Configuración de troncales VoIP con plantillas para ciertos proveedores

IM

OpenFire	Interfaz embebida para administrar el servidor mensajería instantánea llamado Openfire
----------	--

Reports

CDR Report	Reporte de detalles de llamadas (CDRs) con opciones de filtrado
Channel Usage	Reporte de uso de canales muestra gráficos por diferentes tipos de tecnología como SIP e IAX
Billing	Especificación de preferencias de tarificación
à Rates	Establecer tarifas dependiendo de la troncal y del prefijo telefónico
à Billing Report	Tarificación histórica con opciones de filtrado.
à Destination Distribution	Visualización de gráfico de pastel de la distribución para destinos por costo, duración y número de llamadas
à Billing Setup	Configuración de las troncales habilitadas para la tarificación y tarifa por defecto
Asterisk Logs	Reporte del sistema de Asterisk (<i>log</i>) con opciones de filtrado
Graphic Report	Visualización de datos generales de llamadas entrantes y salientes en formato gráfico para extensiones, troncales y colas
Summary by Extension	Resumen de llamadas entrantes, salientes y su duración por cada extensión con reporte gráfico

Extras

vTigerCRM	Permite integración de Elastix con funcionalidad de CRM
Calling Cards	Brinda funcionalidad de tarjetas de llamadas vía A2Billing
Downloads	Listado de descargas recomendadas
à Softphones	Aplicaciones de <i>softphones</i> disponibles
à Fax Utilities	Aplicaciones de fax disponibles
à Instant Messaging	Aplicaciones de mensajería instantánea (IM) disponibles

Addons

Available	Listado de módulos disponibles para extender la funcionalidad de Elastix
Installed	Listado de módulos instalados

6.9 Administración de usuarios

Como ya vimos antes en este mismo capítulo, Elastix crea por omisión un usuario llamado “admin” que posee privilegios de administrador del sistema. Este usuario puede parecer todo lo que necesitamos en un principio pero Elastix nos da la posibilidad de crear nuevos usuarios con distintos niveles de privilegios, permitiéndonos de este modo que usuarios diferentes al administrador accedan a la interfaz pero con privilegios limitados.

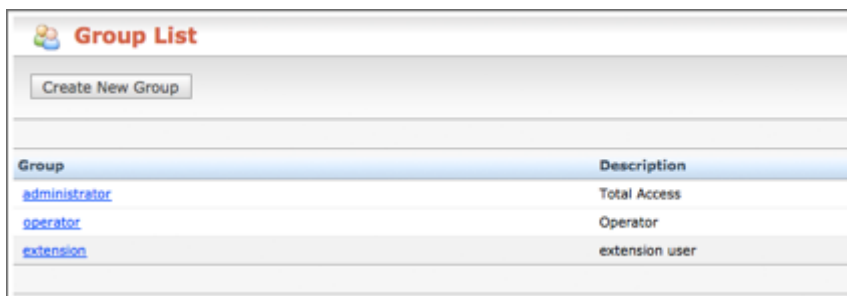
Por ejemplo, podríamos crearle un usuario al recepcionista telefónico para que pueda tener acceso al panel de operador y contestar, parquear o transferir llamadas de una manera amigable. También podríamos crearle un usuario al departamento de auditoría para que tenga acceso a diferentes reportes como el detalle de llamadas y pueda ver el consumo telefónico hacia destinos internacionales. Podríamos también crearle un usuario al departamento de ventas para que puedan tener acceso al CRM integrado.

En general Elastix nos proporciona un poderoso y flexible sistema de permisos que nos permitirá crear un usuario inclusive a cada empleado de la empresa usuaria de Elastix.

Grupos de usuarios

Todo usuario de la interfaz Web de Elastix debe pertenecer a un grupo. Por ejemplo, el usuario “admin” que hemos usado antes pertenece a un grupo llamado “administrator”. Este grupo tiene acceso a todos los menús de Elastix.

Elastix viene con 3 grupos creados por omisión: administrator, operator y extension. Como se puede deducir a simple vista estos grupos sirven para crear usuarios con privilegios de administrador, operador y extensión, respectivamente.



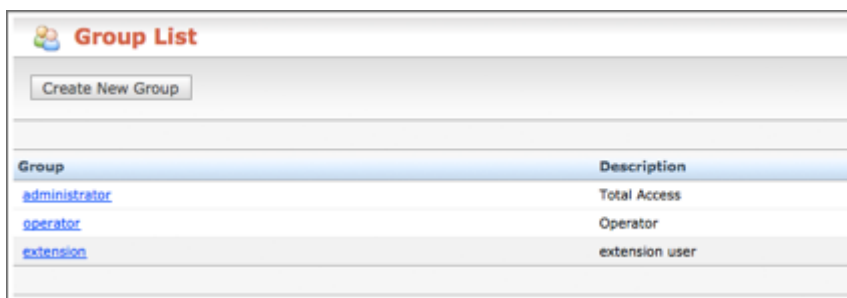
Group	Description
administrator	Total Access
operator	Operator
extension	extension user

Como podemos ver en la figura anterior también es posible crear nuevos grupos de usuarios. Cada grupo de usuarios tiene asociado una matriz de permisos que explicaremos más adelante. Del mismo modo es posible cambiar la matriz de permisos asociada a los grupos creados por omisión. Es por esto que párrafos más arriba hablábamos de un sistema de permisos flexible.

Permisos de grupo

Cada grupo de usuarios tiene una matriz de permisos que asocia el grupo de usuarios con los módulos a los cuales dicho grupo tendrá acceso. De este modo podemos configurar una matriz personalizada para cada necesidad, únicamente seleccionando los módulos adecuados.

Esta matriz de permisos se configura desde “Sistema à Administrar Usuarios à Permisos de Grupo”.



Group	Description
administrator	Total Access
operator	Operator
extension	extension user

Creación de usuarios

Ahora que ya se han explicado los conceptos de grupos de usuarios y permisos de grupo procederemos a crear un nuevo usuario para la interfaz Web de Elastix.

El módulo de usuarios se encuentra ubicado en “Sistema à Administrar Usuarios à Usuarios”. Allí veremos el listado de usuarios activos así como también un botón con la etiqueta “Crear Nuevo Usuario”. Damos clic en dicho botón y accederemos a la siguiente pantalla.



Nuevo Usuario		
Guardar	Cancelar	
Login: *	Nombre (Ex. John Doe): *	
Contraseña: *	Confirmar Contraseña: *	
Grupo: *	Extensión: *	
* Campo requerido		
Administrador		1000
Perfil de Mail		
Usuario de Webmail:	Dominio de Webmail:	
Contraseña de Webmail:	Reingrese contraseña de Webmail:	

A continuación una breve explicación del significado de cada uno de los campos.

- Login: El nombre de usuario (username). No debe contener espacios en blanco.
- Nombre: El nombre completo del usuario. Por ejemplo “Juan Pérez”.
- Contraseña: La clave de acceso

- **Confirmar Contraseña:** Se debe introducir nuevamente la clave de acceso para verificar.
- **Grupo:** El grupo de usuarios al cual pertenecerá el usuario.
- **Extensión:** La extensión de PBX que se asociará al usuario. De este modo cuando el usuario ingrese a la interfaz podrá ver sus voicemails, grabaciones, detalles de llamadas y otra funcionalidad relacionada con su extensión.
- **Usuario de Webmail:** El nombre de usuario de correo electrónico creado en el propio Elastix.
- **Dominio de Webmail:** El nombre de dominio del servidor de correo electrónico.
- **Contraseña de Webmail:** La clave de la cuenta de correo electrónico.
- **Reingrese contraseña de Webmail:** Se debe introducir nuevamente la clave de la cuenta de correo electrónico para propósitos de verificación.

7 Configuración Básica de PBX

Educación es elevar al hombre al nivel de su tiempo

- **José Martí**

La característica de PBX es una de las más importantes de Elastix. En este capítulo explicaremos cómo configurar una PBX básica usando solamente la interfaz Web de Elastix.

Una vez terminada la configuración de la PBX podremos hacer llamadas entre extensiones y hacer uso de algunas características interesantes como voicemail o música en espera.

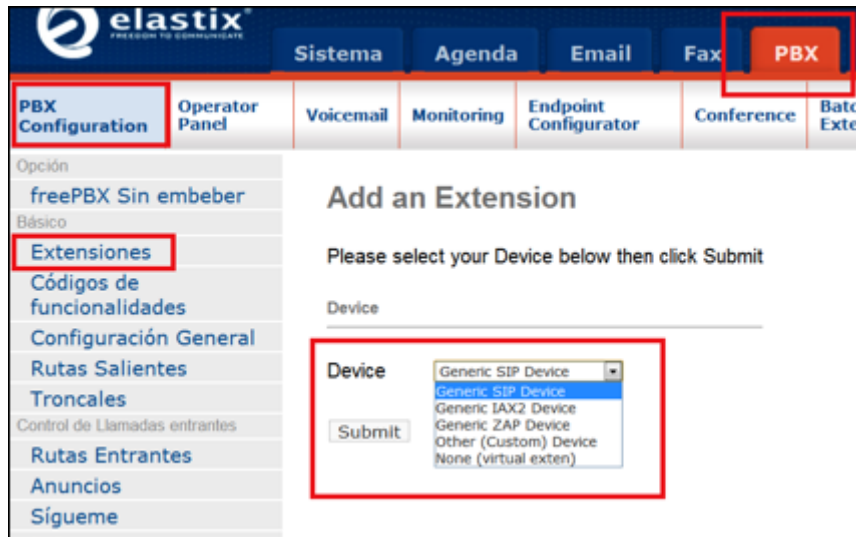
En los siguientes capítulos aprenderemos cómo conectar esta PBX al mundo exterior para poder utilizarla en la práctica.

7.1 Creando una extensión

Definir y editar extensiones es probablemente la tarea más común realizada por un administrador de PBX. En la versión 2.0 de Elastix es posible la creación de cinco tipos de dispositivos o tecnologías soportadas para definir una extensión: SIP, IAX2, DAHDI, "Custom" y Virtual.

Para crear una "Nueva extensión" debemos ingresar al Menú "PBX". Por defecto se accede a la sección "PBX Configuration". 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:



- Generic SIP Device: El SIP es el protocolo estándar para teléfonos IP y ATAs. La mayoría de teléfonos IP soportan SIP.
- Generic IAX2 Device: IAX es el protocolo “Inter Asterisk Exchange”, un nuevo protocolo soportado actualmente por algunos dispositivos.
- Generic ZAP Device: Este dispositivo está asociado a los canales DAHDI. Es un dispositivo de hardware conectado al servidor Elastix.
- 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 relacionar una extensión a un número “externo”. Por ejemplo, para encaminar la extensión 211 a número 1-800-555-1212, se puede crear una extensión “Custom” con número 211 y en la caja de texto del campo “dial” se puede ingresar: Local/18005551212@outbound-allroutes.
- None (virtual exten): Añade la capacidad de crear una extensión virtual.

A continuación 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.

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 sólo los más importantes.

- **User Extension:** Debe ser único. Éste es el número que se puede marcar de cualquier otra extensión. Puede ser de cualquier longitud, pero convencionalmente se utiliza una extensión de tres o cuatro cifras.
- **Display Name:** Es el nombre del Caller ID para que las llamadas de este usuario puedan ser identificadas con un nombre. Sólo debe ingresar el nombre, no la extensión.
- **Secret:** Esta es la contraseña usada por el dispositivo de telefonía para autenticarse contra el servidor de Asterisk que viene en Elastix. Esto es configurado generalmente por el administrador antes de entregar el teléfono al usuario y generalmente no se requiere que lo conozca el usuario. Si el usuario está utilizando un softphone, entonces necesitará saber esta contraseña para configurar su software.

Configuración de teléfono IP

Para probar si la extensión creada hasta aquí funciona correctamente debemos configurar un teléfono IP para que se registre en el servidor Elastix (con las credenciales de acceso creadas). Si el lector desea probar esto ahora puede remitirse al capítulo 9 de la presente obra, donde se explica en extenso cómo configurar tanto un teléfono IP físico como un softphone instalado en un computador.

7.2 Configuraciones generales

En esta sección se configuran parámetros generales del sistema Elastix en lo que tiene que ver con telefonía.

Para acceder a la configuración general debemos ir al menú “PBX à Configuración General”. Allí nos aparecerá la siguiente pantalla.

The screenshot shows the 'Dialing Options' configuration page in Asterisk. It is divided into several sections: 'Dialing Options' with fields for Asterisk Dial and Outbound Dial command options; 'Call Recording' with settings for Extension Recording Override, Call recording format, Recording Location, and Run after record; 'Voicemail' with settings for Ringtime Default, Direct Dial Voicemail Prefix, Direct Dial to Voicemail message type, Optional Voicemail Recording Gain, and a checkbox for 'Do Not Play "please leave message after tone" to caller'; 'Voicemail VmX Locator' with settings for Default Context & Pri, Timeout/#-press default, Loop Exceed default, Timeout VM Msg, Max Loop VM Msg, Direct VM Option, Msg Timeout, Msg Play, and Error Re-tries; 'Company Directory' with settings for Find users in the Company Directory by, Announce Extension, and Operator Extension; 'International Settings' with Country Indications and 24-hour format; 'Security Settings' with Allow Anonymous Inbound SIP Calls; and 'Online Updates' with Check for Updates and Update Email. A 'Submit Changes' button is at the bottom.

Formulario de modificación de configuraciones generales

A continuación se explicarán las opciones de dicha pantalla.

Opciones para “Company Directory”

- **“Find users in the Company Directory by”**: El directorio de una empresa se puede escuchar marcando *411 ó # cuando se escucha un IVR. Esta opción configura de qué modo se ubicarán los usuarios en el directorio. Existen 3 maneras: deletreando las 3 primeras letras del apellido, deletreando las 3 primeras letras del nombre, y deletreando las 3 primeras letras del apellido y nombre al mismo tiempo.

- **“Announce Extension”**: Activa el mensaje que reproduce: “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. Esto es útil en casos donde el llamante no conoce previamente la extensión a la cual se le va a transferir, como por ejemplo cuando marca una opción (Ej. Opción 3) para ser transferido a un representante de ventas. En estos casos puede ser útil que el llamante sepa la extensión a la que fue transferido para poder recordarla en futuras llamadas y saltarse el IVR.
- **“Operator Extension”**: Aquí se debe ingresar la extensión del operador, la cual será el destino si un llamante presiona el cero “0”. Éste destino podría ser una extensión, un grupo de extensiones o inclusive un número telefónico externo.
- **“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 “yes”.

Opciones del comando Dial

Tal como se observa en la figura anterior la mayoría de ocasiones este campo se llena con las letras “tr”. Realmente aquí se han ingresado dos opciones, la opción “t” y la opción “r”, lo que significa que la persona que recibe la llamada puede transferirla y que adicionalmente se enviará un tono de timbrado de vuelta al llamante.

A continuación se muestran todas las opciones disponibles.

- **A(X)**: Toca un anuncio a la persona llamada, usando X como archivo.
- **D(dígitos)**: Envía la secuencia de dígitos DTMF luego que el destino ha contestado.
- **h**: Permite a la persona llamada colgar enviando el dígito “*”.
- **H**: Permite al llamante colgar enviando el dígito “*”.
- **r**: Envía un tono de timbrado a la parte llamante.
- **T/t**: Permite a la parte (llamante/llamada) transferir llamadas.
- **W/w**: Permitir a la parte (llamante/llamada) iniciar la grabación de la llamada en curso, el código por defecto para iniciar la grabación de llamada es “*1”.

7.3 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 creadas 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 cómo 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:

- **Sonar todos**: Timbra las extensiones de todos los agentes hasta que uno conteste. Esta es la opción por defecto.
- **Agente con última llamada más antigua**: Timbra la extensión del agente que menos

recientemente contestó una llamada de esta cola.

- Agente con menos llamadas: Timbra la extensión del agente que menos llamadas haya atendido en esta cola.
- Aleatorio: Timbra las extensiones de manera aleatoria.
- Con memoria: Timbra la extensión de cada agente de manera secuencial, pero recuerda cuál fue el agente que atendió la última llamada para proceder a timbrar la próxima llamada al siguiente agente.
- Linear: Timbra las extensiones de los agentes en el orden especificado, ó en el orden en que los agentes se han logoneado a la cola.
- Wrandom: Timbra las extensiones de los agentes de manera aleatoria pero utilizando una penalidad como medida de probabilidad.

Para configurar una cola en Elastix tenemos que ir al Módulo PBX y escoger en el menú llamado "Colas".

Para añadir una nueva cola deberemos llenar el siguiente formulario.

- **Número de Cola:** Este es el número que debe ser marcado para ingresar a la cola.
- **Nombre de Cola:** Un nombre para la cola. Este es usado únicamente para propósitos de identificación.
- **Contraseña de Cola:** A una cola se le puede poner una clave. Cualquier llamante que intente ingresar a la cola debe digitar dicha clave.
- **Prefijo del nombre del CID:** Como un agente puede iniciar sesión y atender llamadas de varias colas al mismo tiempo, es útil colocarle un prefijo al Caller ID de las llamadas para que cuando el agente reciba la llamada entrante en su teléfono, pueda saber a través de este CallerID que la llamada pertenece a una cola determinada.
- **Agentes Fijos:** Estos son las extensiones que siempre atenderán llamadas que lleguen a la cola.

- **Agentes Dinámicos:** Aquí se especifican los miembros que no usan una extensión fija para atender las llamadas de la cola, sino que deben iniciar sesión en la cola y atienden desde cualquier extensión que esté disponible.
- **Anuncio de Agente:** Este es un anuncio que es reproducido al agente antes de que una llamada le sea transferida. Esto es útil cuando los agentes no tienen detección de Caller ID en su teléfono y se desea saber que una llamada proviene de una cola específica.
- **Clase de Música en espera:** Hace posible la selección de qué clase de música será reproducida al llamante mientras está en la cola a la espera de que un agente lo atienda.
- **Tiempo máximo de espera:** Es el tiempo máximo que una llamada permanecerá en la cola antes de ser desviada la llamada al destino en caso de fallo. La acción a tomar después de este tiempo es configurada en la sección "Destino en caso de fallo".
- **Llamantes Máximos:** Es el máximo número de personas permitidas a esperar en la cola. Si este número es alcanzado, las llamadas en exceso serán desviadas al destino configurado en "Destino en caso de Fallas".
- **Entrar si vacía:** Si se desea que los llamantes ingresen a una cola vacía (sin agentes), entonces hay que cambiar este parámetro a "yes". Esto no es recomendado.
- **Salir cuando vacía:** Se cambia a "yes" si se desea no dejar ingresar más llamadas en el momento en que haya salido el último agente de atención en la cola.
- **Estrategia de timbrado:** Hay 7 tipos de estrategia de timbrado, las cuales fueron explicadas anteriormente:

1. - Sonar todos
2. - Agente con última llamada más antigua
3. - Agente con menos llamadas
4. - Aleatorio
5. - Con memoria
6. - Linear
7. - Wrandom

- **Tiempo de espera de agente:** El tiempo en que le sonará la extensión a cada agente miembro de la cola antes de enviar la llamada a otro agente.
- **Reintentar:** El tiempo que el sistema va a esperar luego de haber hecho sonar todas las extensiones de todos los agentes miembros antes de volver a llamarlos a todos.
- **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.
- **Grabación de llamadas:** Opción de grabar las llamadas entrantes a la cola.
- **Destino en caso de falla:** Este es el destino si una llamada de la cola no pudo ser atendida.

7.4 IVR y sistema de grabación

Grabación de mensaje de bienvenida

Antes de Crear un IVR es necesario tener un mensaje ya grabado que será reproducido cuando una llamada es atendida por dicho IVR. Usted tiene 2 opciones para este fin: grabar el mensaje desde su propia extensión o cargar un archivo de audio grabado anteriormente.

La primera opción sin duda es la más cómoda puesto que podemos grabar desde el mismo teléfono en un tiempo muy corto. Sin embargo la segunda opción nos da la posibilidad de cargar un archivo de audio producido en estudio. Existen compañías que se dedican a brindar estos servicios de grabación de voces para IVRs y dependiendo de nuestro presupuesto pueden incluir voces profesionales, música

de fondo, efectos de sonido, entre otras características.

Para acceder a este módulo diríjase al Menú “PBX”, aparecerá por defecto la sección “Configuración PBX”, en el panel izquierdo escoja la opción “Grabaciones del sistema”.



Como ya se mencionó la primera opción nos permite grabar el audio directamente desde una extensión ya funcional. Para esto ingresamos el número de extensión desde el cual queremos realizar la grabación en el recuadro que dice “Si desea realizar y comprobar grabaciones desde su teléfono, por favor, escriba aquí su extensión”, en este caso usaremos la extensión 201 y damos clic sobre el botón Ir.

Luego de hacer esto, Asterisk estará esperando nuestra grabación en la extensión 201, para continuar descolgamos el teléfono de la extensión ingresada y marcamos *77, se escuchara un pitido el cual indicará que podemos comenzar a vocalizar nuestro mensaje, para indicarle al sistema que hemos concluido el mensaje presionamos la tecla # (numeral), caso contrario no se grabará el mensaje.

Para revisar nuestra grabación desde el mismo teléfono marcamos *99, a continuación comenzará a reproducir el mensaje que acabamos de grabar. Si usted se encuentra conforme con el resultado, el siguiente paso será asignar un nombre a la grabación en el recuadro correspondiente y luego dar clic sobre el botón “Guardar”.

La segunda opción que tenemos es cargar una grabación creada en otro medio, para ello debemos tener un archivo de audio con un formato soportado por Asterisk. En el mismo menú anterior haga clic sobre el botón “Examinar...”, buscamos nuestro archivo, luego procedemos a darle un nombre a esta grabación. Finalmente damos clic en “Guardar”.

Configurar un IVR de bienvenida

El IVR nos permite recibir una llamada entrante y reproducir un mensaje de bienvenida. Además podemos tener un menú interactivo por teclado telefónico, a través de los 10 dígitos, y los símbolos #

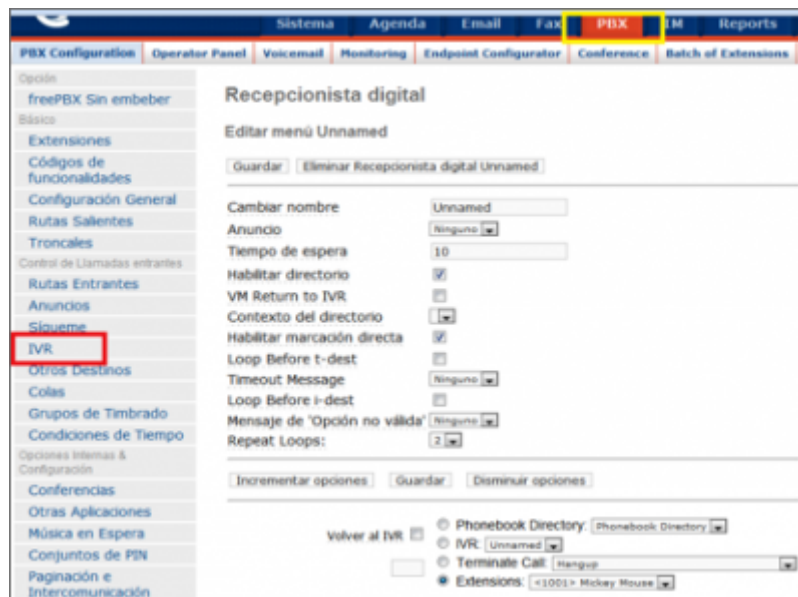
(numeral) y * (asterisco). Con esto es posible encaminar la llamada hacia otro destino o nuevamente hacia el IVR que envió el anuncio.

- Para acceder al módulo “IVR” dirijase al Menú “PBX” y aparecerá por defecto la sección “Configuración PBX”. En el panel izquierdo escoja la opción “IVR”.
- Para grabar un mensaje de bienvenida, como ya se explicó en la sección anterior, debe dirigirse al menú de “Grabaciones del sistema” y proceder a la grabación de un mensaje.
- Para la creación de un IVR no es necesario llenar todos los campos, pues para nuestro caso (un IVR de bienvenida), no necesitaremos opciones.

Los campos necesarios son los siguientes:

- Cambiar nombre: Etiquetar el IVR con un nombre distintivo.
- Tiempo de espera: Tiempo de espera (en segundos) antes de encaminar la llamada a un operador después de haber sido reproducido el mensaje pregrabado.
- Habilitar marcación directa: Opción que permite a quien llama marcar una extensión interna directamente sin tener que esperar a que el operador conteste.
- Anuncio: Es el anuncio o mensaje de bienvenida que se grabó anteriormente. Aparecerá una lista con todos los mensajes disponibles.

A continuación se debe proceder a configurar cuáles serán los destinos de las llamadas de acuerdo con el dígito que el llamante presione en su teléfono. Por ejemplo, generalmente el dígito 0 (cero) hace que la llamada sea enviada a la extensión del operador. Es posible configurar destinos para los números del 0 al 9.



Entre las opciones del menú disponibles, en la parte izquierda existe un casillero donde se debe ingresar el dígito correspondiente a la opción del IVR. Luego debe escoger cuál será el destino de la llamada. Aquí usted podrá seleccionar entre varios módulos disponibles en FreePBX, ya sea enviar la llamada a un cola de atención ya definida, a otro IVR, a un grupo de timbrado, a un extensión específica, etc.

Finalmente grabamos el IVR haciendo clic en el botón de “Guardar”

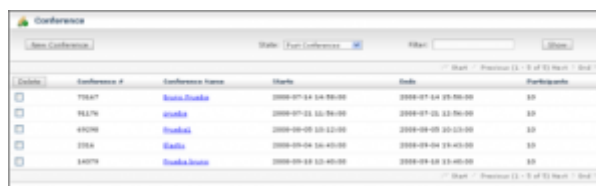
7.5 Salas de conferencias

Elastix puede 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 à PBX Configuration à Conferences”, mientras que el segundo se puede administrar bajo el menú “PBX à Conferencias”.

Los cuartos de conferencia no son otra cosa que una especie de extensión virtual en la que a través de ella se pueden interconectar 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 número 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 #	Conference Name	Start	End	Participants
77637	Conferencia 1	2008-07-04 04:00:00	2008-07-04 05:00:00	0/0
95276	Conferencia 2	2008-07-05 02:00:00	2008-07-05 03:00:00	0/0
49236	Conferencia 3	2008-08-05 00:00:00	2008-08-05 01:00:00	0/0
2764	Conferencia 4	2008-08-04 04:00:00	2008-08-04 05:00:00	0/0
84079	Conferencia 5	2008-08-02 02:00:00	2008-08-02 03:00:00	0/0

En el presente apartado explicaremos cómo administrar conferencias temporales.

Al ingresar al módulo de conferencias se podrá observar todas las conferencias concurrentes. Para visualizar conferencias pasadas o futuras debe seleccionar la opción correspondiente en el recuadro de “Estado” y haciendo clic en “Mostrar”. Además se provee un filtro usado para los nombres de las conferencias.

Para poder eliminar una o varias conferencias debemos seleccionarlas y dar clic sobre el botón “Eliminar”.

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

A continuación se mostrará la manera de administrar las conferencias temporales.

- Para 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 que 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 expulsar de la conferencia a uno o varios invitados seleccionándolos en la columna "Botar" y de clic sobre este botón ("Botar").

Si desea expulsar 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 expulsó todavía 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".

7.6 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 atendemos una llamada y debemos transferirla a una extensión de la que todavía no sabemos el numero destino. Por ejemplo, ingresa una llamada importante para el gerente de la empresa y él no está en su oficina. Dada la importancia 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 extensión del teléfono de recepción impidiendo que nuevas llamadas puedan ser atendidas por la recepcionista. Una vez localizado al esquivo gerente se le puede comunicar que tiene una llamada parqueada en el lote 3434, por ejemplo. Luego, el mismo gerente puede des-parquear dicha llamada y reiniciar la conversación desde su propia extensión o la extensión más cercana.



Habilitar característica de aparcamiento:

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

Extensión de aparcamiento:

Indica la extensión a la cual deberá llamarse para enviar una llamada al parqueo. Generalmente indica el primer slot de canales de parqueo. Dependiendo del número de slots de aparcamientos va a depender cuántas extensiones se tengan para este fin.

Cantidad de aparcamientos:

Indica el número de slots disponibles para parquear llamadas. De éste número depende cuantas extensiones se tengan disponibles para aparcar una llamada.

Tiempo máximo de aparcamiento:

El período de tiempo que una llamada puede permanecer parqueada. Si una llamada cumple ese tiempo en el cuarto de parqueo se intentará retornar la llamada a la extensión desde donde se envió el parqueo.

Contexto del aparcamiento:

Este es el contexto que regirá el comportamiento de las llamadas parqueadas. Por defecto este contiene el contexto "parkedcalls" y no debería ser cambiado.

Destination for Orphaned Parked Calls:

Sirve para configurar el destino de una llamada cuando esta no puede ser enviada al parqueo y quien envió originalmente la llamada 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.

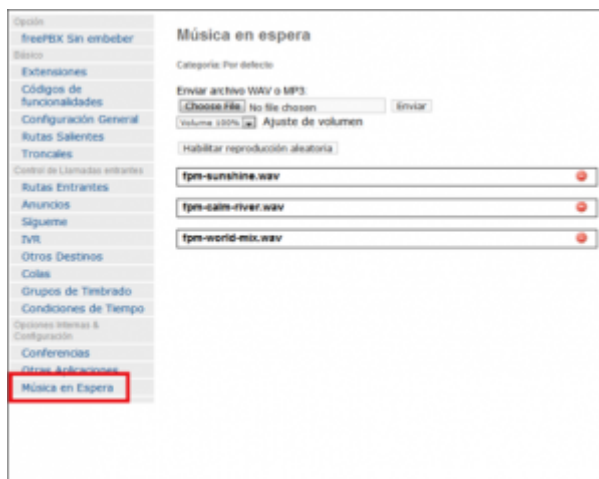
7.7 Música en espera

En esta sección es posible configurar las categorías de "Música en espera" que estarán disponibles para las diferentes colas de atención. El usuario escuchará la reproducción sucesiva de los archivos allí definidos como fondo mientras espera que le atiendan.

Enviar archivo WAV o MP3:

Simplemente mediante el botón "choose file" se selecciona el archivo (en formato mp3 o wav). Luego se debe hacer clic en "Enviar" y debe aparecer el archivo disponible en la lista de archivos de MOH (Music-On-Hold) que se muestran debajo de la opción de cargado de archivo.

Es posible ajustar el volumen de los archivos cargados utilizando la opción "Ajuste de volumen" que se encuentra abajo del botón de "Enviar"



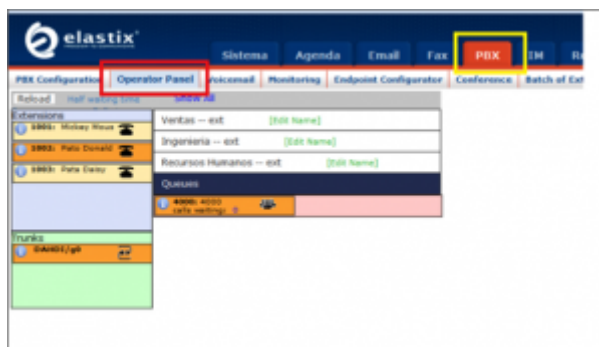
7.8 Panel de Operador

A partir de Elastix 2.0, se ha incorporado el “Elastix Operator Panel” (EOP) que se presenta como una alternativa al existente “Flash Operator Panel” (FOP).

La razón por la cual se introdujo el EOP se debe a que la nueva versión del FOP (FOP2) requiere un tipo de licencia que es incompatible con la actual licencia de Elastix (GPL versión 2).

De manera general el panel de operador permite visualizar el estado de las extensiones, troncales, colas, cuartos de conferencias y cuartos de parqueo del sistema en tiempo real. Es una aplicación muy útil para una recepcionista.

Para acceder al “Elastix Operator Panel” diríjase al Menú “PBX” y haga clic en la utilidad “Operator Panel”.



Adicionalmente, en Elastix 2.0 aún se encuentra disponible el “Flash Operator Panel” en el menú de “PBX” en el submenú “Flash Operator Panel”.



7.9 Funciones de transferencia de llamadas

Por omisión en Elastix viene configurada la funcionalidad de transferencia de llamadas de la siguiente forma:

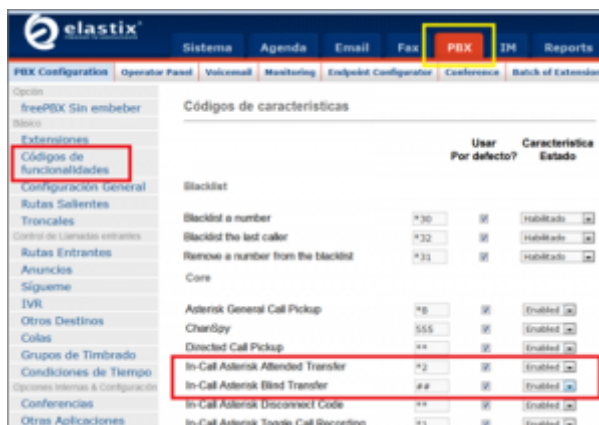
Transferencia ciega

Para transferir 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 éstos códigos es necesario dirigirse al menú de “PBX à PBX Configuration” y escoger el menú de “Códigos de funcionalidades”. En la sección “Core” usted tendrá las opciones de “In-Call Asterisk Attended Transfer” e “In-Call Asterisk Blind Transfer” donde podrá cambiar el código asignado para las transferencias atendidas y ciegas respectivamente.



8 Configuración de hardware

No siempre podemos agrandar, pero siempre podemos tratar de ser agradables

- Voltaire

A pesar de que Elastix puede funcionar sin necesidad de hardware telefónico instalado en el mismo servidor, es necesario su utilización si se requiere conexión directa a ciertos tipos de redes telefónicas como la PSTN. Su instalación y configuración se simplifica con Elastix gracias a que se dispone de una interfaz Web especializada que facilita esta tarea.

En el presente capítulo nos referiremos básicamente a tarjetas PCI (o PCI express). Sin embargo pueden usarse otros dispositivos para conectarse a la PSTN como es el caso de los gateways, que funcionan a través de IP o channel banks que pueden inclusive funcionar a través de USB.

8.1 Tipos de tarjetas y fabricantes

En este apartado nos referiremos básicamente a tarjetas telefónicas, también llamadas tarjetas de interfaz. Este hardware está disponible tanto para redes 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

Existen variadas opciones de hardware compatible con Elastix por lo que no tendremos problemas para encontrar una que se adapte a nuestras necesidades.

Para facilitar todavía más la elección Elastix cuenta con un programa de certificación de hardware que se encarga de someter cada modelo a diversas pruebas para determinar su compatibilidad con Elastix de acuerdo a los criterios del programa de certificación.

La lista de hardware certificado está disponible de manera pública en el siguiente enlace:

<http://www.elastix.org/en/product-information/certified-hardware.html>

Aquí no solo encontraremos tarjetas de interfaz sino también teléfonos IP, gateways, intercomunicadores, channel banks, cámaras de seguridad y appliances.

Cabe resaltar que si un determinado modelo no se encuentra enlistado no necesariamente significa que no es compatible con Elastix. Debido a que el programa de certificación estipula que el proceso de pruebas debe darse únicamente por solicitud explícita del fabricante, puede darse el caso de que el modelo que estamos buscando no haya sido propuesto por su fabricante para ser sometido al proceso de certificación.

8.2 Instalación de hardware de telefonía en Elastix

Instalar hardware de telefonía en Elastix es una tarea relativamente sencilla. Sin embargo, es necesario tener claros los conceptos antes de comenzar pues de lo contrario podríamos confundirnos.

Para instalar hardware de telefonía en Elastix debemos seguir los siguientes pasos.

1. Instalar el hardware físicamente
2. Detectar el hardware a nivel de drivers
3. Configurar el hardware a nivel de Asterisk

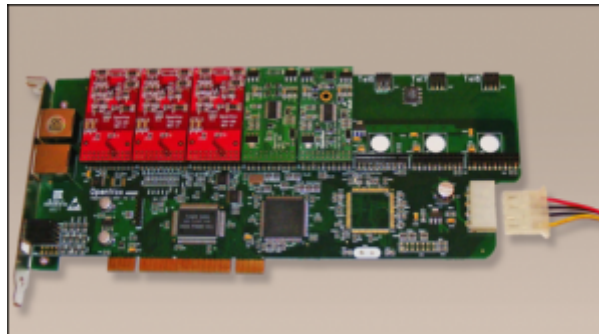
Cada una de estas tareas se explicará a continuación.

8.3 Instalación física de una tarjeta

Esto no involucra ninguna tarea complicada, solo las precauciones obvias al instalar tarjetas electrónicas en un computador, como haber apagado y desconectado completamente el equipo de la

toma eléctrica, usar un dispositivo de descarga electrostática, entre otras.

- 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.



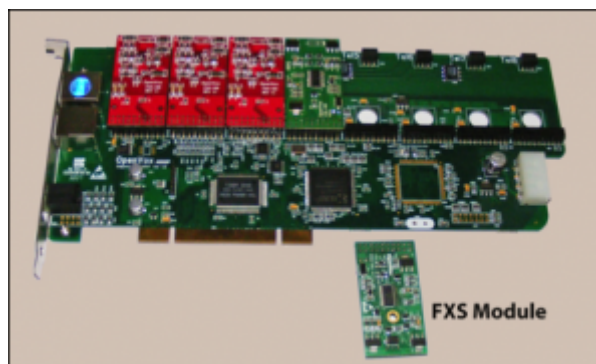
En la figura anterior podemos observar en la parte inferior derecha el conector tipo molex que sirve para proveer de energía adicional a la tarjeta en caso de que usemos módulos FXS.

Luego de instalar la tarjeta encendemos el equipo y deberíamos verla enlistada a nivel de bus PCI. Como ejercicio podríamos utilizar el comando `lspci` antes y después de instalar la tarjeta y ver la diferencia.

Hardware modular

Existen fabricantes que diseñan su hardware de manera modular, de tal forma que pueda ser expandible. Los módulos por tanto no se encuentran soldados sino que pueden ser instalados a presión.

Por ejemplo, demos un vistazo a la tarjeta analógica de la siguiente imagen que tiene capacidad de hasta 8 puertos. Se puede observar claramente los módulos y que el fabricante los ha diferenciado: los módulos FXO son de color rojo y los FXS son de color verde.



Existen también otro tipo de tarjetas donde inclusive el cancelador de eco es un módulo que se puede instalar a presión. De este modo se puede adquirir la tarjeta con o sin cancelador de eco por hardware.

8.4 Detección de hardware

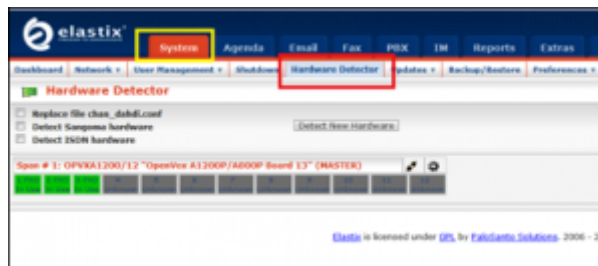
Una vez que la tarjeta ha sido instalada es hora de que los drivers la detecten. El driver de telefonía más importante en Elastix recibe el nombre de DAHDI (realmente se trata de un conjunto de drivers). No entraremos en muchos detalles de DAHDI por ahora puesto que la interfaz Web de administración nos permitirá abstraernos de la labor de los drivers. Sin embargo, más adelante explicaremos un poco más a fondo el funcionamiento de DAHDI y cómo se relaciona con Asterisk.

- Nota: Cabe mencionar que DAHDI no es el único driver de telefonía existente en Elastix. Si bien la mayoría de fabricantes de hardware de telefonía usan DAHDI (o han integrado sus drivers dentro de DAHDI), existen otros fabricantes que proveen drivers diferentes como es el caso de Sangoma o Rhino Equipment. Estos otros drivers también se encuentran integrados dentro de Elastix.

En las primeras versiones de Elastix la tarea de detectar el hardware se hacía manualmente ejecutando algunos comandos desde la consola del equipo. Actualmente se cuenta con una interfaz Web que detecta el hardware de manera automática y nos muestra gráficamente el hardware detectado.

Esta interfaz se puede acceder desde el menú “System → Hardware Detector”.

Para detectar nuevo hardware basta con presionar el botón “Detectar Nuevo Hardware”.



En dicho menú se tienen las siguientes opciones:

- Replace file chan_dahdi.conf”: Esta opción crea un archivo /etc/asterisk/chan_dahdi.conf adicionalmente a los necesarios que se requieren para el sistema DAHDI (/etc/dahdi/modules.conf y /etc/dahdi/system.conf) que será procesado por Asterisk, conteniendo la información de configuración de las tarjetas que se detecten.
- Detect Sangoma Hardware”: Esta opción activa el funcionamiento del demónico de “wanrouter” provisto por el fabricante SANGOMA para reconocimiento de su hardware y su integración con los módulos de DAHDI.
- Detect ISDN hardware”: Esta es una opción heredada de versiones anteriores de Elastix, utilizada exclusivamente para detección de hardware del tipo “mISDN”. Comúnmente se utiliza esta opción para la detección de tarjetas BRI de Digium y sus clónicas.

Para las tarjetas de telefonía analógica y digitales se encuentra disponible la herramienta para configurar el tipo de cancelación de eco (basado en software) que se utilizará para cada puerto de la tarjeta. Para configurar el cancelación de eco haga clic en el ícono que parece una rueda.

A continuación se ingresará a la configuración particular de la tarjeta mostrando cada uno de los

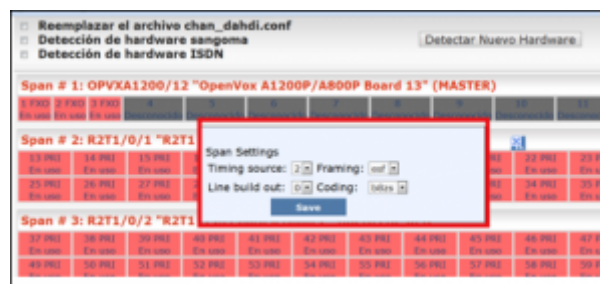
puertos de la misma y un menú desplegable para la elección del algoritmo requerido por cada puerto. Los algoritmos soportados son: OSLEC, MG2, KBL, SEC, SEC2.

- Nota: El algoritmo más usado es OSLEC. Se puede encontrar más explicación acerca de los canceladores de eco y OSLEC en el capítulo 13.



Para las tarjetas de telefonía digital, se encuentra disponible una utilidad para configurar los parámetros de sincronización necesarios y particulares para líneas E1 ó T1 en forma gráfica. Para acceder a dicha utilidad debe hacer clic en el ícono que tiene la forma de dos conectores.

A continuación se ingresará a la configuración particular de la tarjeta con un menú de opciones desplegables. La elección de los valores de las opciones dependerá del tipo de interfaz digital y el servicio que su proveedor de telefonía le haya instalado.



En la sección siguiente se explicarán con más detalles todos los parámetros necesarios para la sincronización para tarjetas digitales.

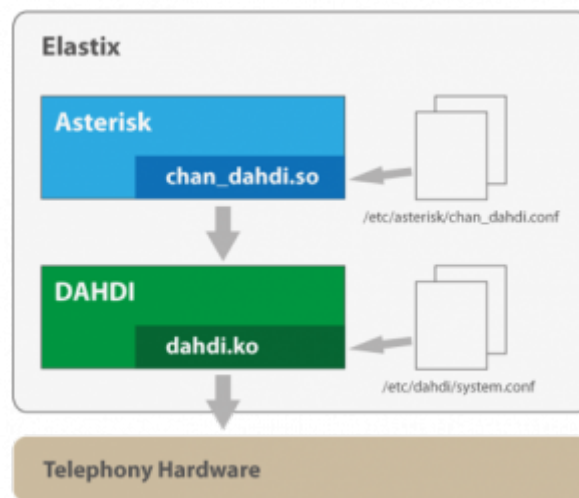
Una vez ejecutada la acción de reconocimiento de hardware, Elastix crea archivos de configuración con la información suministrada y añadiendo información adicional que la considera como "usualmente" correcta, denominada "por defecto".

A partir de ésta configuración por defecto el usuario podría modificar ciertos parámetros más específicos para su ambiente, para que el hardware de telefonía se ajuste a su situación particular. Estas configuraciones a bajo nivel se explicarán en la siguiente sección.

- 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.

8.5 Configuración del hardware de telefonía a nivel de

Asterisk



Podemos ver en el diagrama anterior que Asterisk se relaciona con DAHDI a través de un módulo (módulo de Asterisk) llamado `chan_dahdi.so` y que este módulo necesita del archivo `/etc/asterisk/chan_dahdi.conf` para saber cómo comunicarse con DAHDI correctamente.

- Nota: Más adelante entraremos en detalles acerca del archivo `/etc/dahdi/system.conf` mostrado en el diagrama anterior.

Antes de entrar en detalles de configuración del archivo `chan_dahdi.conf` debemos mencionar que una manera sencilla y automática de configurar nuestro hardware a nivel de Asterisk es presionar el botón "Replace `chan_dahdi.conf`". Como el lector podrá suponer esta acción reemplaza el archivo `chan_dahdi.conf` por uno genérico. Sin embargo, en la mayoría de los casos la configuración escrita en este archivo sólo nos será de utilidad como ejemplo o punto de inicio debido a que Elastix no tiene manera de conocer cómo queremos agrupar los canales telefónicos u otros parámetros que dependen de nuestro ambiente específico y que se configuran en dicho archivo.

A continuación explicaremos cómo configurar adecuadamente el archivo `chan_dahdi.conf`

Configuración del archivo `chan_dahdi.conf`

Asterisk cuenta con un módulo llamado `chan_dahdi.so` que sirve para comunicarse con los drivers de DAHDI y de esta manera conectar Asterisk con hardware telefónico. Este módulo se configura a través del archivo `chan_dahdi.conf`

En el archivo `chan_dahdi.conf` básicamente se configura la señalización de los canales de voz y se relacionan los canales de Asterisk con los circuitos o líneas telefónicas definidos en la configuración de los drivers. Estos canales se pueden agrupar en lo que se conoce como troncales.

- Nota: La configuración de los drivers de DAHDI reside principalmente en el archivo `/etc/dahdi/system.conf`. Este archivo se escribió automáticamente cuando realizamos la detección de hardware desde la interfaz Web.

Es decir que si tengo 15 circuitos o líneas telefónicas que terminan en hardware telefónico de mi

servidor los puedo controlar desde 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 `chan_dahdi.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-pstn
channel=1-5

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

; trunk #2
group=2
language=es
callprogress=no
context=from-pstn
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`. A continuación la tabla de posibles valores para este parámetro.

```
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: FX0 (Loop Start)
fxo_gs: FX0 (Ground Start)
fxo_ks: FX0 (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 (FX0 at the
channel bank)
fxs_tx: Transmit audio/PTT on an FXS loopstart interface (FX0 at the
channel bank)
fxo_rx: Receive audio/COR on an FX0 loopstart interface (FXS at the
channel bank)
fxo_tx: Transmit audio/PTT on an FX0 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_trx: Receive audio/COR AND Transmit audio/PTT on an E&M interface (2-
way)
em_rxtx: same as em_trx (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_trx: Receive audio/COR AND Transmit audio/PTT on an SF interface (2-
```

```
way)
sf_rxtx: 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
nil: ISDN nacional tipo 1
dms100: Nortel DMS100
4ess: AT&T 4ESS
5ess: Lucent 5ESS
euroisdn: EuroISDN
qsig: Protocolo usado en ISDN
```

8.6 Configuración de interfaces digitales utilizando los archivos de configuración DAHDI

La herramienta de detección de hardware basada en Web que incorpora 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 asumiendo parámetros por defecto que pueden ser personalizados de acuerdo al ambiente real de su servidor.

Los cambios necesitan ser hechos directamente en los archivos de configuración, por lo tanto se requerirá tener conocimiento más profundo a este nivel.

- Nota: En este apartado nos referiremos básicamente a circuitos **T-Carrier** y **E-Carrier** basados en los drivers de DAHDI.

Configuración de drivers DAHDI

El hardware de telefonía que instalemos en el servidor Elastix debe ser reconocido en primera instancia por los drivers (Ej.: DAHDI) y luego ser activado en la configuración de Asterisk.

DAHDI lee su configuración principalmente del archivo

```
/etc/dahdi/system.conf.
```

Para interfaces digitales, en el archivo **/etc/dahdi/system.conf** se deben definir los spans.

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 E1 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 span (telco) como fuente de sincronización. DAHDI proveerá la sincronización y la enviará al otro lado. Esto es más común cuando se conecta a través de este span una PBX antigua a la cual hay que proveerle sincronización. Normalmente si nos conectamos a una compañía telefónica no deberíamos tener ningún span con este valor de timing
1	Usar el span (telco) como fuente primaria de sincronización
2	Usar el span (telco) como fuente secundaria de sincronización
3,4,5,.....,n	Usar el span (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 - 40 metros
1	40 - 81 metros
2	81 - 121 metros
3	121 - 162 metros
4	162 - 200 metros
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).

Después de realizar los cambios en el archivo **system.conf** será necesario reiniciar el demonio de DAHDI. Si todo ha sido correctamente configurado, DAHDI cuenta con el comando "lsdahdi", el cual al ser ejecutado desde la consola del sistema operativo mostrará las tarjetas configuradas y el respectivo estado de sus canales.

Si a pesar de que todo se encuentre correctamente configurado y se reciban señales de alarma o errores debe asegurarse que no exista algún problema con el cableado físico o que el proveedor de la troncal digital tenga un problema en su lado.

8.7 Detalles de DAHDI

En esta sección se detallan los nombres de los archivos importantes que vienen como parte del sistema DAHDI y una breve descripción de cada uno de ellos.

Archivos de configuración DAHDI

Archivos DAHDI	Descripción
/etc/dahdi/system.conf	Archivo de configuración donde se define la señalización de cada uno de los puertos de telefonía.
/etc/asterisk/chan_dahdi.conf	Este archivo realmente es un archivo de configuración de Asterisk que se utiliza para decirle a Asterisk con qué canales DAHDI se debe comunicar y cómo.
/etc/dahdi/init.conf	Este es un archivo utilizado por el script de inicio de DAHDI (/etc/init.d/dahdi).
/etc/dahdi/modules	Este archivo contiene los nombres de los drivers de DAHDI que se encuentran presentes en nuestro sistema. (recordemos que DAHDI está compuesto por varios drivers). Este archivo es leído por el script de inicio de DAHDI para saber qué drivers debe cargar y de este modo no cargar todos los drivers sino solo los necesarios.

Módulos de kernel de DAHDI

Módulo DAHDI en Kernel	Descripción
dahdi.ko	Módulo de Kernel (driver) que contiene la funcionalidad principal de DAHDI

Módulo DAHDI en Kernel	Descripción
dahdi_dummy.ko	Driver necesario para proveer de una fuente de timing a Asterisk cuando no tenemos tarjetas instaladas, debido a que las tarjetas poseen una fuente de timing vía hardware

Módulos de Asterisk de DAHDI

Módulo DAHDI en Asterisk	Descripción
chan_dahdi.so	Módulo de Asterisk que se comunica con los drivers de DAHDI
app_dahdibarge.so	Módulo de Asterisk que provee la aplicación DAHDIBarge a nivel de plan de marcado. Esta aplicación sirve para monitorear (escuchar) un canal de DAHDI.
app_dahdiscan.so	Módulo de Asterisk que provee la aplicación DAHDIScan a nivel de plan de marcado. Esta aplicación sirve para navegar entre canales y monitorearlos.

Comandos DAHDI

Comando DAHDI	Descripción
dahdi_cfg	Configura el hardware a partir del archivo de configuración /etc/dahdi/system.conf
dahdi_monitor	Monitorea los niveles de entrada y salida de un canal DAHDI
dahdi_scan	Imprime la configuración de los spans DAHDI
dahdi_speed	Realiza pruebas de desempeño sobre hardware DAHDI
dahdi_test	Prueba si el temporizador de DAHDI está funcionando correctamente
dahdi_tool	Muestra el estado de las interfases DAHDI. Muy útil para hacer depuración.
dahdi_genconf	Genera de manera automática archivos de configuración para el hardware DAHDI

8.8 Antes de DAHDI: equipos que usaban Zaptel

A partir de Elastix 1.5 el paquete Zaptel fue reemplazado por el paquete DAHDI. Debido a que es un cambio relativamente reciente es probable que en la práctica nos encontremos con estos equipos que todavía usan Zaptel. Para que podamos configurar adecuadamente dichos equipos en esta sección presentamos un resumen de las diferencias más importantes entre estos dos paquetes de drivers.

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 son importantes tener en cuenta y que se describen a continuación.

Cambios en archivos de configuración

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

Cambios de nombre de módulos de kernel

Módulo DAHDI	Módulo Zaptel
dahdi.ko	zaptel.ko
dahdi_dummy.ko	ztdummy.ko

Cambios de nombre de módulos de Asterisk

Módulo DAHDI	Módulo Zaptel
chan_dahdi.so	chan_zap.so
app_dahdibarge.so	app_zapbarge.so
app_dahdiscan.so	app_zapscan.so

Cambios de nombre de algunos comandos

comando DAHDI	Comando Zaptel
dahdi_cfg	ztcfg
dahdi_monitor	ztmonitor
dahdi_monitor	ztmonitor
dahdi_scan	ztscan
dahdi_speed	ztspeed
dahdi_test	zttest
dahdi_tool	zttool
dahdi_genconf	zapconf o genzaptelconf

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 el IP del servidor Elastix), el nombre de usuario (comúnmente la extensión) y la contraseña de registro 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 tenga la seguridad que la configuración de teléfonos soportando protocolo IAX no difiere mucho de lo aquí mostrado. De forma general, en este libro se usará el protocolo SIP en la mayoría de los ejemplos con excepción de los casos donde los teléfonos no se encuentren conectados dentro de la red local, en cuyo caso será preferible la utilización del protocolo IAX, ya que no es susceptible a problemas relacionados con la traslación de direcciones IP (NAT).

Al final del capítulo se detallará la manera de configurar una gran cantidad de teléfonos IP de manera ágil y sencilla. Este método nos permite pre-configurar los teléfonos desde una herramienta llamada "endpoint configurator" disponible en Elastix.

9.1 Configuración de softphone

Actualmente el uso de teléfonos en software o softphones se ha popularizado en gran medida más que nada por el hecho del ahorro que puede significar al no tener que adquirir teléfonos físicos. El ahorro es mayor mientras más extensiones se requieran, por lo que ésta alternativa es muy común en call centers.

Por supuesto antes de empezar a configurar el softphone habrá que crear una extensión desde Elastix.

A manera de ejemplo vamos a suponer que se ha creado la extensión 5678 la cual utiliza tecnología SIP como protocolo VoIP y su "secret" asignado es "8765".

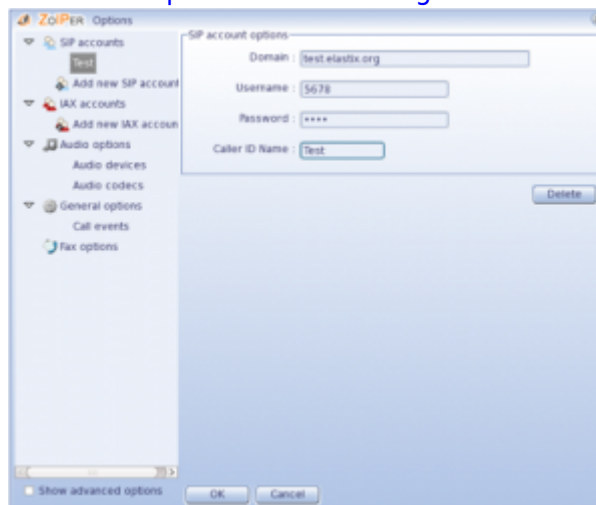
El softphone que vamos a configurar será el **Zoiper** que está disponible en versión gratuita y comercial. La versión gratuita es suficiente para que podamos hablar cómodamente de un punto a otro así que basaremos nuestro ejemplo en esta versión.

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.



A continuación presionamos el botón con la figura de herramienta y accederemos al menú de opciones. Se desplegará 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.

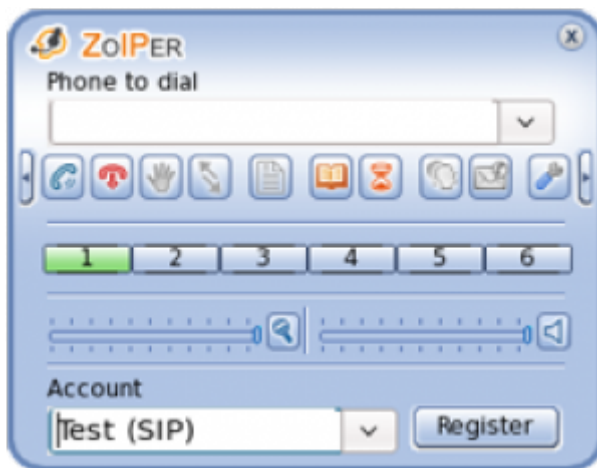


Llenaremos dicha pantalla con los siguientes datos:

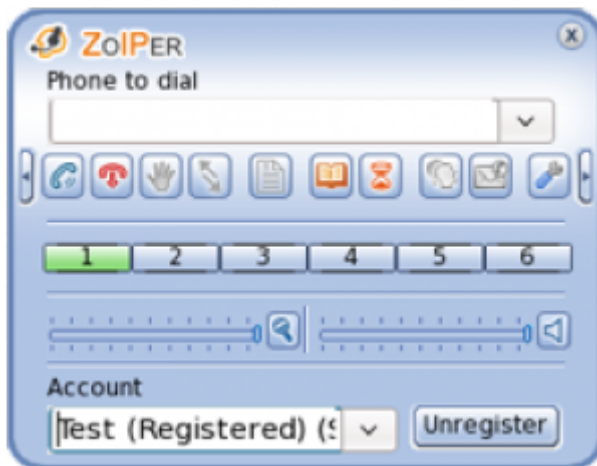
Domain: La dirección IP o nombre de dominio del servidor Elastix.
Username: 5678

Password: 8765
Caller ID Name: El nombre que aparecerá en los teléfonos que reciban nuestras llamadas

Luego de presionar el botón "OK", volvemos a la pantalla inicial del Zoiper. Para proceder a registrar la extensión recientemente configurada en la central Elastix, debemos seleccionar la extensión creada en el combo de la parte inferior y dar clic en el botón "Register".



Si Zoiper se registra correctamente observaremos que junto al nombre de nuestra cuenta "Test" aparecerá la palabra "Registered".



Puede repetir este proceso para configurar otro softphone Zoiper en otra PC utilizando una extensión diferente creada en el mismo Elastix y luego comprobar que es posible establecer llamadas VoIP entre estas dos extensiones. En el ejemplo la extensión 5678 recibe la llamada de la extensión "2222".



- Nota: Asterisk ofrece la posibilidad de comprobar el funcionamiento de extensiones recientemente creadas a través de una extensión de "ECO". Esta extensión contestará

automáticamente y reproducirá el mismo audio que enviemos.

9.2 Configuración de teléfono IP

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 a manera de ejemplo un teléfono Polycom SoundPoint IP 330.



Configuración de red

La configuración de red puede realizarse considerando que por defecto el teléfono obtendrá su IP mediante DHCP, pero también es posible configurar un dirección IP fija. Explicaremos los dos métodos.

Después de la configuración IP del equipo, se accede a la interfaz Web de administración del Teléfono para completar el proceso configurando los parámetros restantes.

DHCP

Mediante esta opción, el teléfono obtendrá una dirección IP de manera automática. Debe tomar en cuenta que la red debe contar con un servidor DHCP funcional, que es quién asignará en su la dirección IP.

Elastix trae disponible un servidor DHCP embebido que se puede activar fácilmente desde la interfaz Web.

A continuación se detalla cómo verificar en el teléfono IP cuál fue la dirección IP que le fue asignada.

Presionar **“Menu à Status à Network à TCP/IP Parameters”**

En la pantalla LCD se mostrará la dirección IP. Anotémosla para luego poder acceder al teléfono desde la interfaz Web.

IP estática

Si usted necesita 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, debe utilizar direcciones IP estáticas en sus teléfonos. Para configurar una dirección IP en el teléfono, deberá ingresar los parámetros de IP, máscara de red, gateway y DNSs en un menú diferente. A continuación las instrucciones de cómo ingresar cada uno de ellos.

Dirección IP

Dado que por defecto el teléfono viene configurado para recibir su dirección IP vía DHCP, a continuación mostramos la manera de cambiar dicho comportamiento a uso de IP fijo. En la interfaz del teléfono Presionar **“Menu à Settings à Advanced à (ingresar clave. Por omisión es 456) à Admin Settings à Network Configuration à DHCP Client à Editar à Disabled à 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” à (Ingresar la máscara de red)**

Puerta de enlace

En el caso de que su teléfono se necesite conectar en un segmento de red IP diferente al que se encuentra conectado el servidor Elastix, el teléfono deberá contar con la dirección IP de la puerta de enlace (por lo general es un router) correspondiente, de lo contrario su teléfono no podrá registrarse en la central Elastix.

Seleccionar **“IP Gateway” à (Ingresar la dirección IP de la puerta de enlace)**

Después de cualquier cambio debemos seleccionar **“Exit” y luego “Save Config”**. Debemos

esperar a que el teléfono termine de reiniciarse.

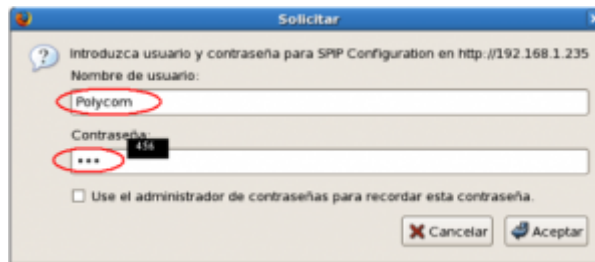
Configuración de registro contra el servidor Elastix

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

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



Antes de ingresar a la interfaz de configuración Web un cuadro de diálogo nos solicitará el usuario y la clave del teléfono. El usuario y clave de fábrica son **"Polycom"** y **"456"** respectivamente.



IP Servidor Elastix

Para ingresar los parámetros de registro contra el servidor Elastix, comenzaremos ingresando los parámetros que se muestran en la pestaña "SIP".

En esta pantalla se podrán configurar una parte de los parámetros **SIP**, como lo son el **servidor SIP**, es decir la **dirección IP de la central Elastix** y 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 192.168.1.160

The screenshot shows the 'SIP Configuration Parameters' page in the Polycom SoundPoint IP Configuration web interface. The 'Servers' tab is selected. The 'Outbound Proxy' section has 'Address' set to '192.168.1.160' and 'Port' set to '5060'. The 'Server 1' section also has 'Address' set to '192.168.1.160' and 'Port' set to '5060'. The 'Transport' dropdown menu is set to 'DNSSnapttr'. A note on the left indicates 'Puerto de registro para la extensión' and a note on the right indicates 'Dirección IP del Elastix'.

Debemos tomar en cuenta de que la dirección de la central Elastix puede pertenecer o no a nuestro segmento de red IP, puesto que la central se encuentra en el Internet, igual será posible llegar a ella a través de la puerta de enlace configurada.

En esta misma pantalla también es posible cambiar el puerto utilizado para señalización SIP contra la central Elastix. Si usted cambia a otro valor diferente al de omisión (5060 UDP) debe asegurarse que este cambio también haya sido hecho en la central Elastix.

Extensión

Para ingresar los parámetros específicos a la extensión asignada al teléfono IP, seleccione la pestaña de "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 interfaz Web del Elastix.

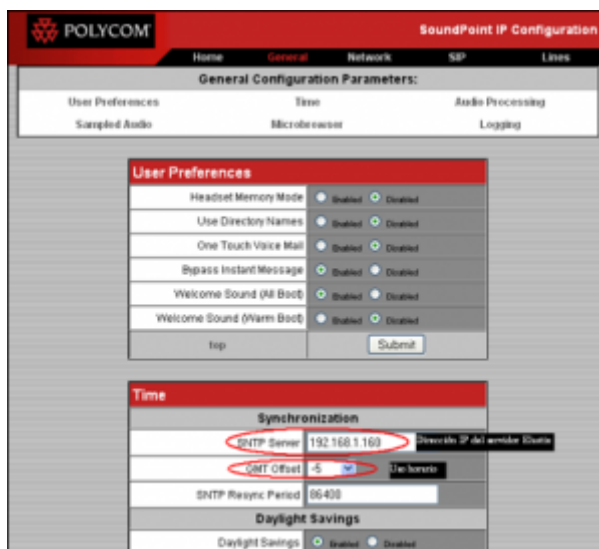


Configuración Opcional

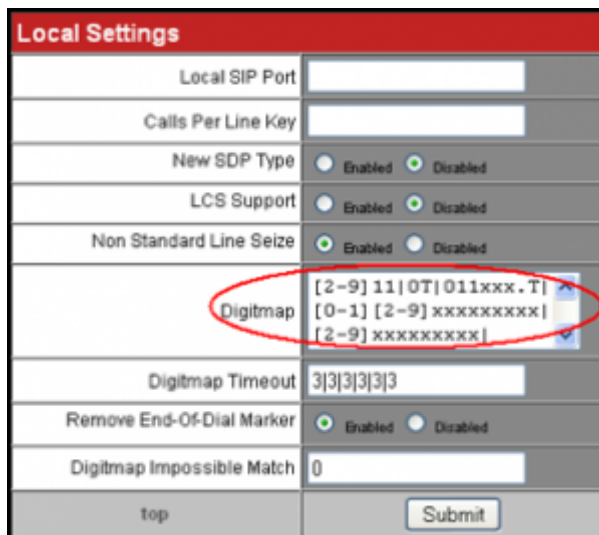
En esta sección veremos cómo configurar el Polycom para que muestre la fecha y hora del Elastix.

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**”



- **Nota:** Si al intentar digitar alguna de las extensiones disponibles en la central Elastix (desde el teléfono que estamos configurando) escuchamos una señal de ocupado aún cuando ni siquiera se han terminado de digitar todos los dígitos de la extensión destino, revise la opción “Digitmap”, dentro de la sección “Local Settings” dentro del menú “SIP”, o en su defecto elimínela por completo.

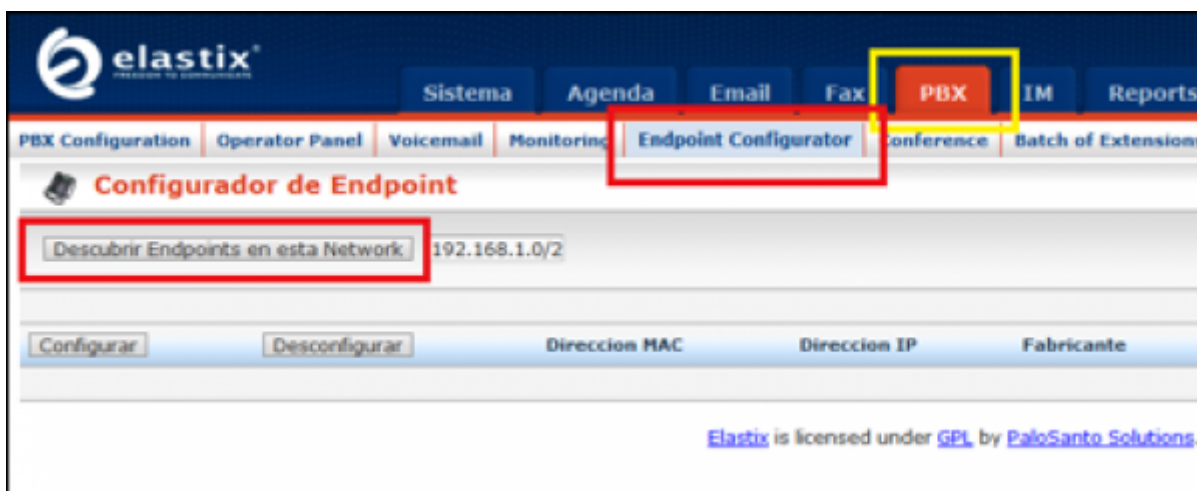


9.3 Utilizando el Endpoint Configurator

En ocasiones la instalación de teléfonos IP se complica cuando son muchos y más aún cuando la configuración no es tan trivial como la que se realiza para el **softphone**. Como ya se mostró en la sección anterior algunos modelos de teléfonos se pueden configurar desde una interfaz Web embebida en el propio teléfono y también existen otros que se tienen que configurar a través de una sesión telnet.

La distro Elastix cuenta con una interfaz que nos permitirá configurar muchos teléfonos de una manera más sencilla y ágil. Esta herramienta recibe el nombre de **Endpoint Configurator**.

El módulo Endpoint Configurator primero 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 que se encuentra configurada en la propia central Elastix; el scan que se ejecuta sobre dicho segmento de red obtiene internamente información de MAC e IP de los dispositivos de red conectados.

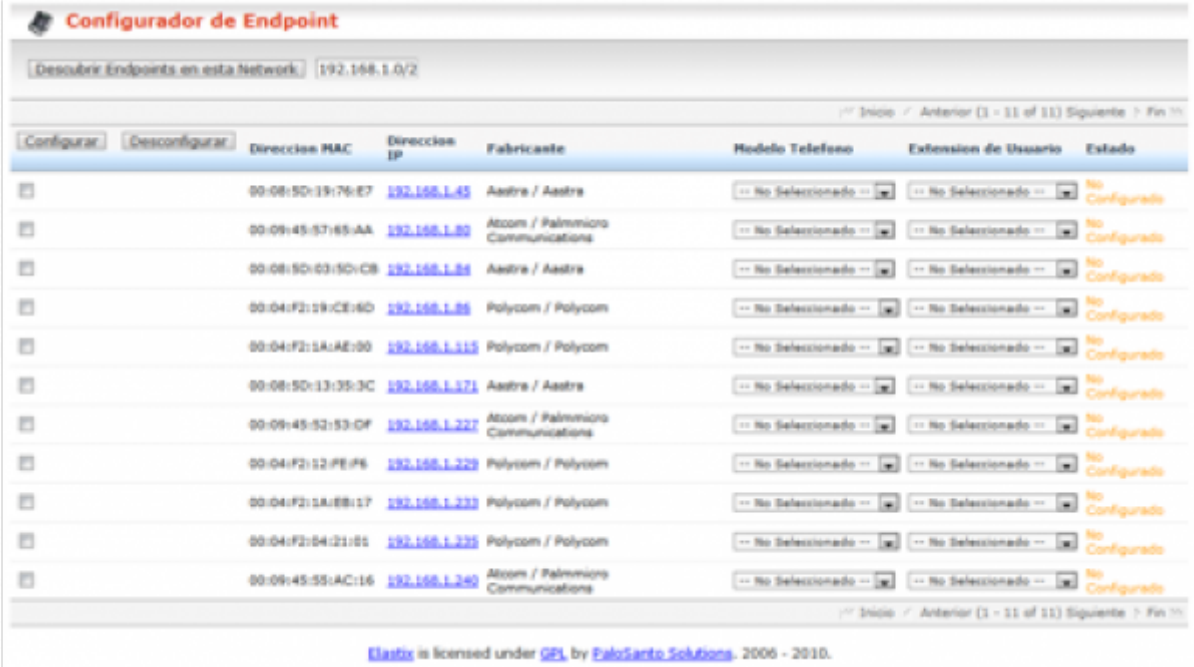


Nota: La dirección MAC le sirve al endpoint configurator como identificador único del teléfono. Si se usara la dirección IP ésta podría cambiar con el tiempo, pero con la dirección MAC no ocurre esto.

Luego de obtener la información de todos los endpoints del segmento de red, el módulo de forma inteligente trata de identificar los equipos que pudieran ser teléfonos IP y de la misma forma trata de

recabar información acerca del fabricante y modelo para elegir cuáles puede configurar consultando su base de datos interna de modelos de teléfonos.

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



The screenshot shows the 'Configurador de Endpoint' interface. At the top, there is a search bar with the text 'Descubrir Endpoints en esta Network' and the IP address '192.168.1.0/2'. Below this, there are navigation buttons for 'Configurar' and 'Desconfigurar'. The main part of the interface is a table with the following columns: 'Dirección MAC', 'Dirección IP', 'Fabricante', 'Modelo Telefono', 'Extensión de Usuario', and 'Estado'. The table contains 10 rows of data, each representing a discovered endpoint. The 'Estado' column for all entries is 'No Configurado'. At the bottom of the interface, there is a footer that reads 'Elastix is licensed under GPL by FaloSanto Solutions. 2006 - 2010.'

	Dirección MAC	Dirección IP	Fabricante	Modelo Telefono	Extensión de Usuario	Estado
<input type="checkbox"/>	00:08:5D:19:76:E7	192.168.1.45	Aastra / Aastra	-- No Seleccionado --	-- No Seleccionado --	No Configurado
<input type="checkbox"/>	00:09:45:57:65:AA	192.168.1.80	Atcom / Palmtronics Communications	-- No Seleccionado --	-- No Seleccionado --	No Configurado
<input type="checkbox"/>	00:08:5D:03:5D:C8	192.168.1.86	Aastra / Aastra	-- No Seleccionado --	-- No Seleccionado --	No Configurado
<input type="checkbox"/>	00:04:F2:19:CE:6D	192.168.1.88	Polycorn / Polycorn	-- No Seleccionado --	-- No Seleccionado --	No Configurado
<input type="checkbox"/>	00:04:F2:1A:AE:00	192.168.1.115	Polycorn / Polycorn	-- No Seleccionado --	-- No Seleccionado --	No Configurado
<input type="checkbox"/>	00:08:5D:13:3D:3C	192.168.1.171	Aastra / Aastra	-- No Seleccionado --	-- No Seleccionado --	No Configurado
<input type="checkbox"/>	00:09:45:52:53:0F	192.168.1.227	Atcom / Palmtronics Communications	-- No Seleccionado --	-- No Seleccionado --	No Configurado
<input type="checkbox"/>	00:04:F2:12:FE:F6	192.168.1.229	Polycorn / Polycorn	-- No Seleccionado --	-- No Seleccionado --	No Configurado
<input type="checkbox"/>	00:04:F2:1A:EB:17	192.168.1.233	Polycorn / Polycorn	-- No Seleccionado --	-- No Seleccionado --	No Configurado
<input type="checkbox"/>	00:04:F2:04:21:01	192.168.1.235	Polycorn / Polycorn	-- No Seleccionado --	-- No Seleccionado --	No Configurado
<input type="checkbox"/>	00:09:45:55:AC:16	192.168.1.240	Atcom / Palmtronics Communications	-- No Seleccionado --	-- No Seleccionado --	No Configurado

Como se puede observar en la figura anterior se muestra 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.

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

Direccion MAC	Direccion IP	Fabricante
00:08:5D:19:76:E7	192.168.1.45	Aastra / Aastra
00:09:45:57:65:AA	192.168.1.80	Atcom / Palmmicro Communications
00:08:5D:03:5D:CB	192.168.1.84	Aastra / Aastra
00:04:F2:19:CE:6D	192.168.1.86	Polycom / Polycom
00:04:F2:1A:AE:00	192.168.1.115	Polycom / Polycom
00:08:5D:13:35:3C	192.168.1.171	Aastra / Aastra

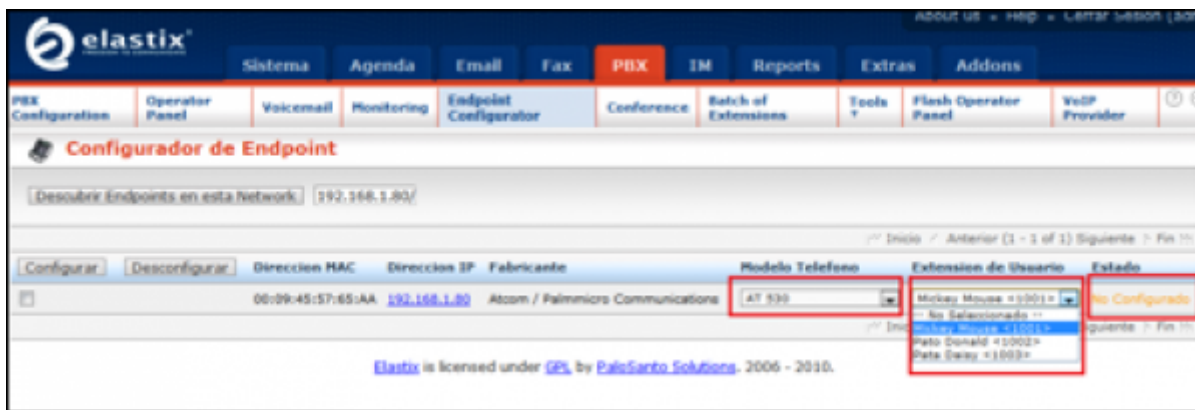
Existen dos columnas llamadas **Modelo de Teléfono y Extension** de Usuario. Estas columnas sirven para asociar una extensión (previamente creada en Elastix) con un teléfono físico. En la lista llamada "Modelo de Teléfono" podrá elegir el modelo del teléfono que se desea asociar con la extensión y solo 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 à PBX Configuration**".

- Nota: Una limitación de esta herramienta es que las extensiones deben ser SIP y no están soportadas extensiones de ningún otro protocolo. Actualmente el grupo de desarrollo de Elastix está trabajando en incorporar soporte para otros protocolos.

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 "No Configurado" lo que significa que todavía la central Elastix no tiene configuración acerca del él.

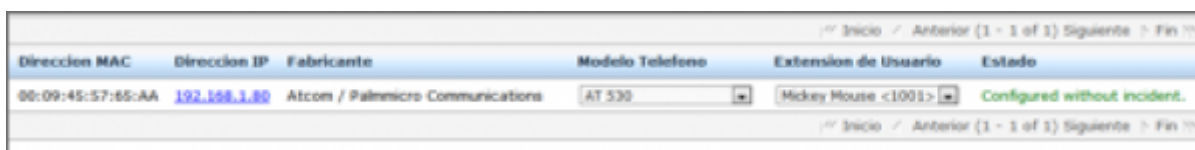


Debe elegir la casilla ubicada en el lado izquierdo de la pantalla, y luego presionar el botón “Configurar”. Inmediatamente aparece la ventana de confirmación en la cual si se está seguro se acepta y el proceso de configuración empezará. Esto puede tomar algo de tiempo dependiendo del número de teléfonos que haya elegido, en este caso por ser un ejemplo hay solo un teléfono a configurar.



Después de este 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 atc0009455765aa.cfg donde 0009455765aa 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 archivo y reiniciar. Por ello usted podrá notar que el teléfono se va a reiniciar.



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.

Vovemos a interpretar los Status del teléfono:

Estatus	Color	Descripción
No Configurado	Naranja	No configurado aun por la central Elastix
Configured without incident	Verde	La central configuró el teléfono sin problemas
UPDATE: Secrets no equals in FreePBX and Endpoint	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 “Desconfigurar”**. La acción que ejecuta éste es sólo a nivel del módulo. La central Elastix borra el registro histórico relacionado con el teléfono IP de su base de datos interna y borra los archivos de configuración relacionados en la ruta **/tftpboot/**. Esto indica que el teléfono va a mantener la configuración que le fue entregada desde la última actualización desde esta herramienta.

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 “No Configurado”.



En la parte izquierda vamos a elegir la casilla, y luego presionamos el botón “Configurar”. 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.



Después 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 realizó varias acciones.

- **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 se detallan en la tabla abajo mostrada.

Archivo	Descripción
0004f21aeb17.cfg	El 0004f21aeb17 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
0004f21aeb17reg.cfg	Contiene información de la (o las) extensiones que va a poseer el teléfono
Polycom	Realmente es una carpeta que posee a su vez otras carpetas llamadas contacts, logs y overrides
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 ejecutar lo siguiente desde el propio teléfono Polycom:
 - * Elegir que el tipo de aprovisionamiento 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 el teléfono IP ya está configurado desde la central Elastix y cuando se desee cambiar de extensión sólo debemos ir al módulo Endpoint Configurator, hacer un par de clics y listo.

Actualización por TFTP

TFTP es el protocolo usado en este caso para proveer aprovisionamiento. Significa **Trivial FTP** y es un protocolo que no requiere de ningún tipo de autenticación, es decir que no necesitamos ni un usuario o contraseña para realizar algún proceso. La mayoría de teléfonos IP soportan 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 descargar la lista de contactos y escribir logs.

La central Elastix trae embebido el servicio encargado de escuchar y aceptar transferencias utilizando protocolo 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. Es posible que en versiones superiores a Elastix 2.0 se cuente con una herramienta basada en Web para realizar estas tareas, pero por ahora el lector podrá hacerlo editando los archivos de 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	En este archivo se encuentra la configuración para algunos modelos ATCOM. Para cambiar alguna configuración se recomienda revisar manuales de los archivos de configuración para ATCOM
Polycom.cfg.php	En este archivo se encuentra la configuración 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	En este archivo se encuentra la configuración para algunos modelos Linksys. Hay que notar que también tienen formato XML
Cisco.cfg.php,	En este archivo se encuentra la configuración para los modelos de teléfonos Cisco soportados
Aastra.cfg.php	En este archivo se encuentra la configuración para los modelos de teléfonos Aastra soportados
Grandstream.cfg.php	En este archivo se encuentra la configuración para los modelos de teléfonos Grandstream soportados
Snom.cfg.php	En este archivo se encuentra la configuración para los modelos de teléfonos Snom soportados

Al hacer 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 Configuración avanzada de PBX

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 fijarse en el DID, en el CallerID ó el número de puerto por donde está recibándose dicha llamada en el caso de que se cuente con troncales tipo

DAHDI..

The screenshot displays the Elastix web interface. At the top, there is a navigation bar with tabs for 'Sistema', 'Agenda', 'Email', 'Fax', 'PBX', 'IM', and 'Reports'. Below this is a sub-navigation bar with 'PBX Configuration' highlighted. A sidebar on the left contains a tree view of configuration options, with 'Rutas Entrantes' (Incoming Routes) selected and highlighted. The main content area is titled 'Add Incoming Route' and contains the following fields and options:

- Add Incoming Route:** A text input field.
- Description:** A text input field.
- DID Number:** A text input field.
- Caller ID Number:** A text input field.
- CID Priority Route:** A checkbox.
- Options:** A section header for the following fields:
 - Alert Info:** A text input field.
 - CID name prefix:** A text input field.
 - Music On Hold:** A dropdown menu with 'Default' selected.
 - Signal RINGING:** A checkbox.
 - Pause Before Answer:** A text input field.
- Privacy:** A section header for the following field:
 - Privacy Manager:** A dropdown menu with 'No' selected.
- Language:** A section header for the following field:
 - Language:** A text input field.
- Fax Detect:** A section header for the following field:
 - Detect Faxes:** Radio buttons for 'No' (selected) and 'Yes'.
- CID Lookup Source:** A section header for the following field:
 - Source:** A dropdown menu with 'None' selected.
- Set Destination:** A section header for the following fields:
 - Phonebook Directory:** A dropdown menu with 'Phonebook Directory' selected.
 - Queues:** A dropdown menu with 'Sales <4000>' selected.
 - IVR:** A dropdown menu with 'Unnamed' selected.
 - Terminate Call:** A dropdown menu with 'Hangup' selected.
 - Extensions:** A dropdown menu with '<1001> Mickey Mouse' selected.
- Buttons:** 'Submit' and 'Clear Destination & Submit'.

Description:

Etiqueta que identifica la ruta entrante.

DID Number:

Para un peer del tipo SIP ó IAX, el DID (Destination-Inward-Dial) es el número destino de la llamada, es asignado por el proveedor y generalmente está asociado al número de cuenta. Por ejemplo si su cuenta es '85679432', ingresando ése número en el campo DID, las llamadas entrantes destinadas a esa cuenta deberían coincidir con esta información y dirigirse al destino configurado. Si deja este campo en blanco se estará haciendo referencia a las llamadas entrantes con cualquier número como destino.

Caller ID Number:

El CID Number es el Caller ID o número telefónico desde donde se origina la llamada que es recibido por Elastix. Este dato no es confiable ya que puede ser fácilmente burlado por hackers. Al dejar este campo en blanco se estará haciendo referencia a las llamadas entrantes con cualquier Caller ID.

Alert Info:

Puede ser usado para activar la característica de rings distintivos soportada en algunos de dispositivos SIP. Requiere de conocimientos particulares de la marca y modelo del dispositivo SIP.

Pause Before Answer:

En esta opción se define el número de segundos que deberíamos esperar antes de contestar la llamada entrante. El primer propósito para esta opción es permitir que dispositivos conectados en paralelo en esta línea, como lo pueden ser máquinas de fax o dispositivos de seguridad electrónica puedan timbrar primero antes de que la central procese la llamada entrante.

Privacy Manager:

Si se activa esta opción, toda llamada que no llegue con un Caller ID válido no podrá ser enrutada al destino final de la llamada si antes no se ingresa manualmente el Caller ID, es decir, al llamante se le solicitará que utilizando su teclado del teléfono ingrese el número de origen.

Language:

Mediante este campo se configura el idioma de los mensajes que serán reproducidos para el tratamiento de la llamada. Escriba "es" para el español.

Detect Faxes:

Si se habilita esta opción Asterisk tratará de reconocer un tono de fax en la llamada, si reconoce un tono de fax, la llamada será enviada a la extensión por defecto de fax que se configura en

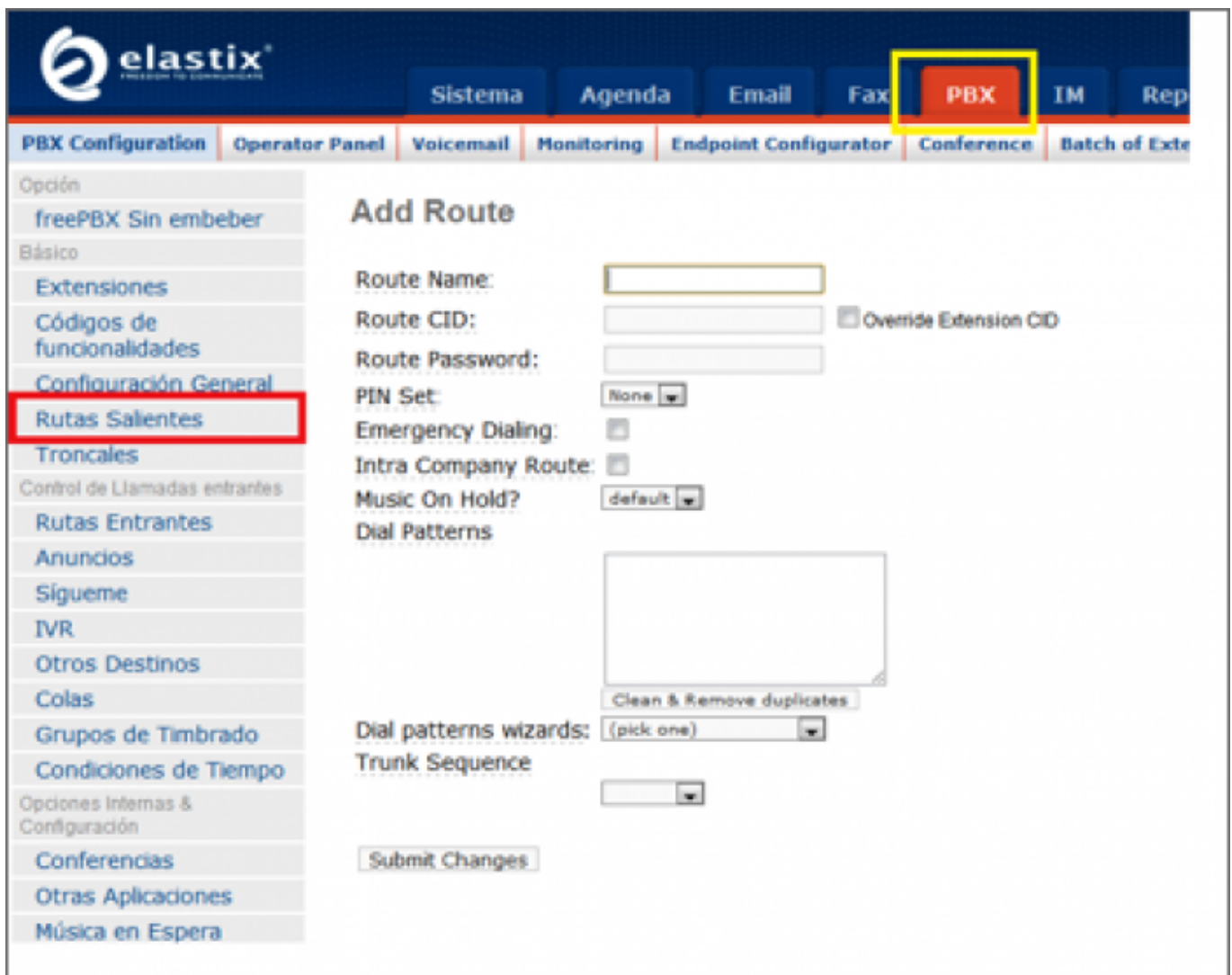
“Configuración General”.

Set destination:

Mediante el “set destination” podemos especificar dónde 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 que originan desde alguna extensión interna de la central Elastix.



Route Name:

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

Route CID:

En este campo es posible especificar un número que será utilizado por la llamada como número de origen. Esta característica de cambiar el número origen a discreción del administrador de la central debe estar soportada por el proveedor de las troncales utilizadas para realizar la llamada.

Route Password:

Si llenamos este campo, cada vez que una llamada intente utilizar esta troncal, al llamador le será solicitada que ingrese ese mismo patrón como clave. Si la clave es ingresada correctamente la llamada es conectada, caso contrario la llamada es descartada. Solo es permitido el ingreso de dígitos.

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 ó patrón de dígitos que Asterisk usa para discriminar si el número marcado por la extensión interna "hace juego" con el patrón configurado en la ruta saliente y de este modo 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:

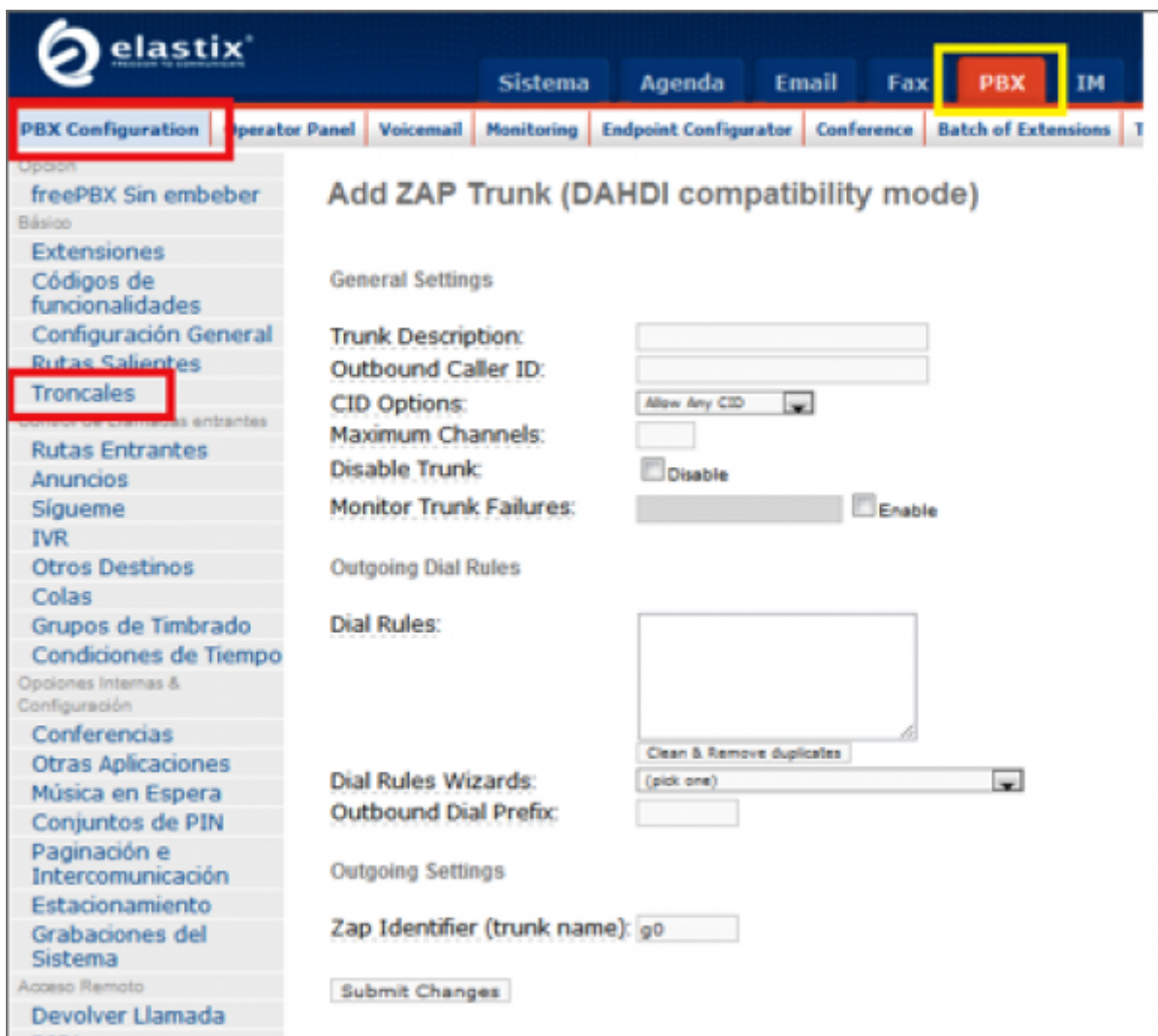
Se lista e indica el orden en que se debe intentar el uso de las troncales. Cuando un número digitado por un llamador "hace juego" con el patrón de marcado especificado, Asterisk intentará realizar la llamada por las troncales especificadas en esta opción en el orden en que son listadas.

10.2 Troncales

Una troncal es aquella que permite terminar una llamada a otro sistema externo de comunicación VoIP. Alguno de los tipos de troncales son:

DAHDI
IAX2
SIP
ENUM
DUNDi
Custom

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



The screenshot shows the Elastix web interface for configuring a ZAP Trunk. The 'PBX' menu item is highlighted in yellow, and the 'PBX Configuration' menu item is highlighted in red. The 'Troncales' menu item in the left sidebar is also highlighted in red. The main content area displays the 'Add ZAP Trunk (DAHDI compatibility mode)' form with the following fields:

- Trunk Description:
- Outbound Caller ID:
- CID Options:
- Maximum Channels:
- Disable Trunk: Disable
- Monitor Trunk Failures: Enable
- Dial Rules:
- Dial Rules Wizards:
- Outbound Dial Prefix:
- Outgoing Settings
- Zap Identifier (trunk name):

A 'Submit Changes' button is located at the bottom of the form.

Trunk Description:

Con este campo se le asigna un nombre distintivo a la troncal.

Outbound Caller ID:

En esta opción es posible especificar el número de origen o Caller ID que la llamada usará cuando use esa troncal. Es necesario que el proveedor de la línea telefónica conectada a ese puerto DAHDI permita el cambio de CallerID.

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 cómo 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 "hace juego" con los 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 mostradas, se deben especificar configuraciones entrantes y

salientes que generalmente son provistas 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=myspassword
sendrpid=yes
type=peer
username=myusername
```

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

```
myusername:myspassword@host
```

Troncales de tipo DAHDI

Las troncales de tipo DAHDI son las asociadas a hardware de telefonía instalado en la máquina y usa el módulo chan_dahdi.so. Los archivos asociados a este tipo de troncales son el /etc/asterisk/chan_dahdi.conf y los archivos bajo la carpeta /etc/dahdi/

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

Troncales personalizadas.

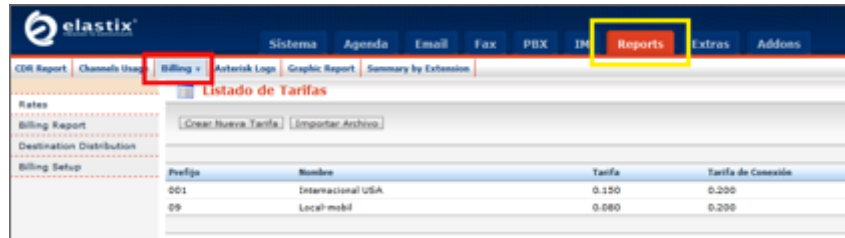
Estas troncales son creadas cuando se usan protocolos no muy comunes en Asterisk como H323, CAPI, etc.

Se debe especificar un string de marcado. Por ejemplo:

```
CAPI/XXXXXXX
```

10.3 Tarifación

La utilidad "Rates" del Menú "Billing" dentro de la pestaña de "Reports" del Elastix nos permite visualizar y configurar las tarifas que se usarán para la facturación de llamadas.



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.



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
- Troncal: Troncal de la cual se tomará las contabilización de las llamadas.

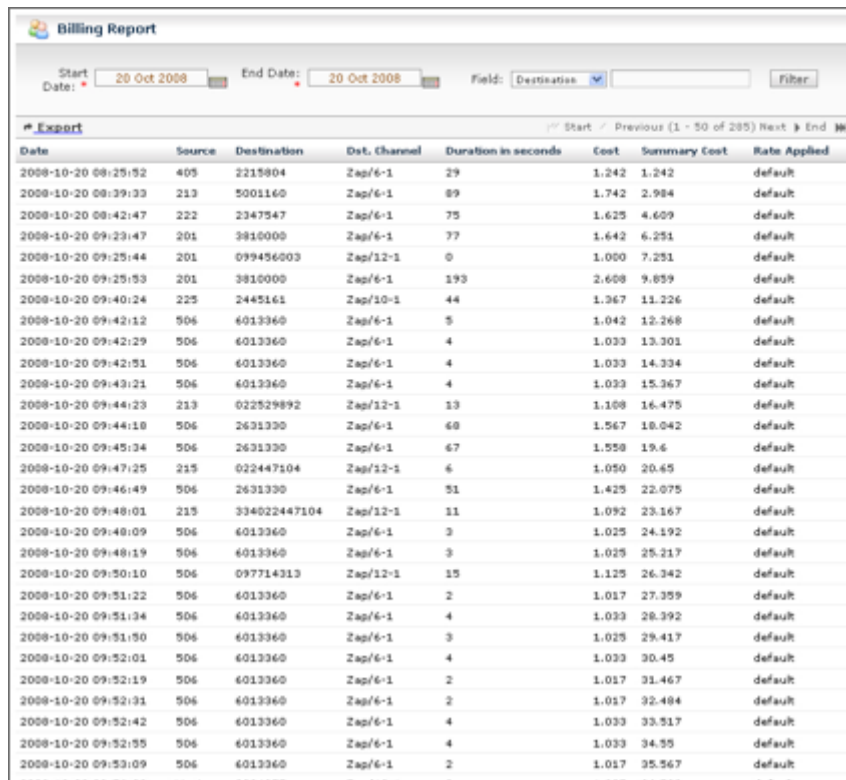


Reporte de Facturación

La opción “Billing Report” del Menú “Billing” 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.



Date	Source	Destination	Dest. 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	1.442	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

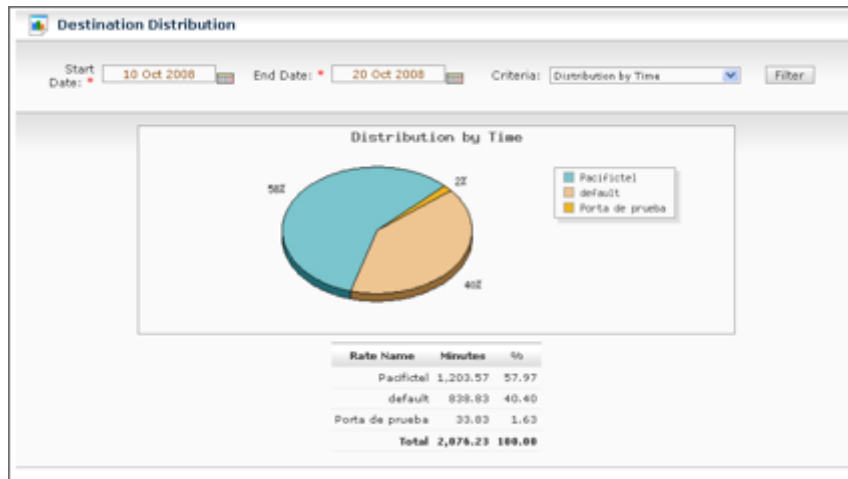
Distribución de Destinos

La opción “Destination Distribution” del Menú “Billing” 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

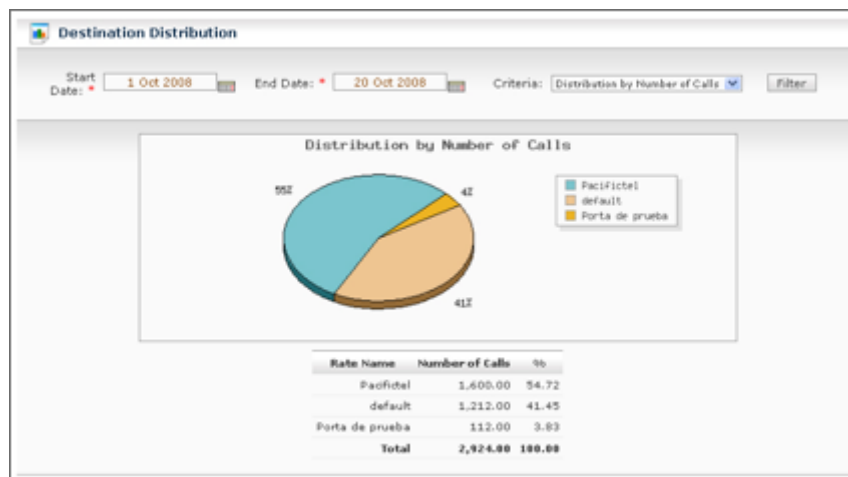
Distribución por Tiempo

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.



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 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 "Billing" 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.



Configuración de Troncales para Facturación

La opción “Configuración de Troncales” del Menú “Billing” 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 click en el botón “Habilitar para Facturar”.



10.4 Reportes

Reporte CDR

CDR es la abreviación de Call Detail Record o Registro de Detalle de Llamada.

La opción “Reporte CDR” en la pestaña de “Reportes” del Elastix nos permite visualizar un listado con el detalle de las llamadas.

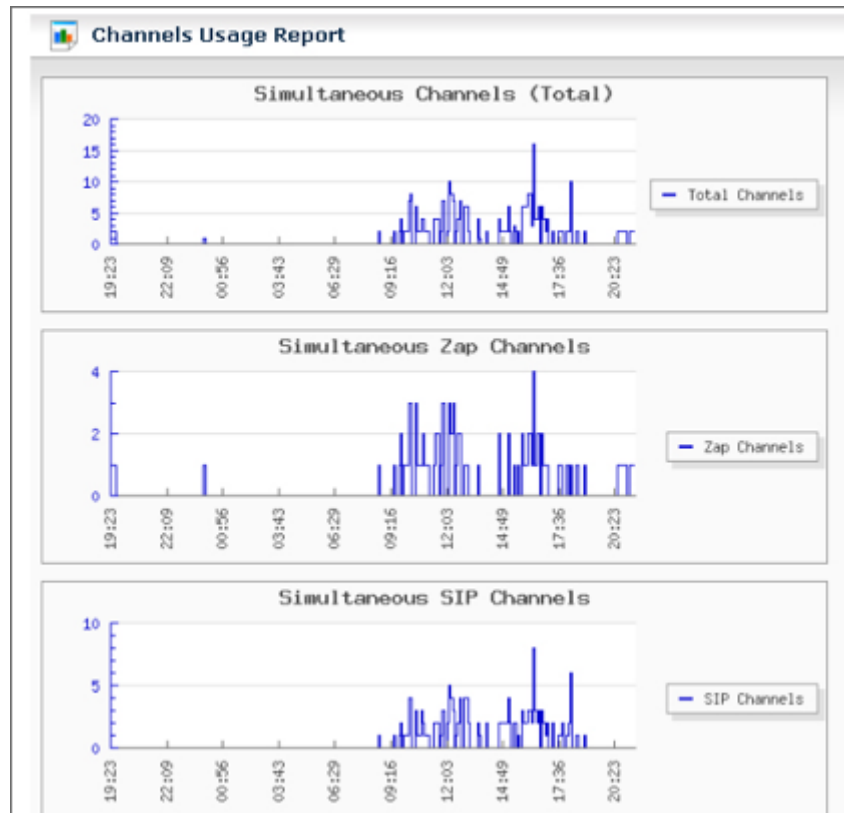
# Export	Date	Source	Destination	Src. Channel	Código de Cuenta	Dest. Channel	Status	Duration
2008-10-13 10:16:43	212	201	Local/201@from-internal-eb35.2	SIP/201-08db940	SIP/201-08db940		NO ANSWER	0
2008-10-13 10:17:32	2000		Zap/10-1	Local/200@from-internal-97b.1			ANSWERED	47
2008-10-13 10:17:49	225	2305000	SIP/225-07d34b18	Zap/7-1			ANSWERED	249
2008-10-13 10:30:33	225	2305000	SIP/225-07d41c20	Zap/7-1			ANSWERED	140
2008-10-13 10:33:45	215	201	SIP/215-07d41c20	SIP/201-08db9570			ANSWERED	20
2008-10-13 10:17:02	216	2517100	SIP/216-07d33580	Zap/6-1			ANSWERED	1193
2008-10-13 10:39:37	225	2403150	SIP/225-07d29610	Zap/6-1			ANSWERED	60
2008-10-13 10:50:59	506	2313103	SIP/506-07d22420	Zap/6-1			ANSWERED	39
2008-10-13 10:49:05	704	201	IAH2/palomasantonia-7212	SIP/201-07b15e00			ANSWERED	103
2008-10-13 10:51:04	215	2275423	SIP/215-07d336c0	Zap/7-1			ANSWERED	74
2008-10-13 10:51:45	506	2313103	SIP/506-07d34c50	Zap/6-1			ANSWERED	87
2008-10-13 10:53:58	502		Local/502@from-internal-96d5.2	SIP/502-08da16d0			NO ANSWER	0
2008-10-13 10:55:07	216	2517100	SIP/216-07d336c0	Zap/7-1			NO ANSWER	0
2008-10-13 10:55:09	216	201	SIP/216-07d291f0	SIP/201-08da16d0			ANSWERED	9
2008-10-13 11:00:23	202	2310305	SIP/201-07d34b20	Zap/7-1			ANSWERED	16
2008-10-13 11:00:47	202	214916	SIP/201-07d06420	Zap/7-1			NO ANSWER	0
2008-10-13 11:00:56	203	2149163	SIP/201-07d079d0	Zap/7-1			ANSWERED	96
2008-10-13 10:58:21	506	2446070	SIP/506-07d305ad	Zap/6-1			ANSWERED	294
2008-10-13 11:03:43	506	2295704	SIP/506-07d29610	Zap/6-1			ANSWERED	89
2008-10-13 11:12:59	203		Local/203@from-internal-334a.2	SIP/203-08d371e0			NO ANSWER	0
2008-10-13 11:12:59	202		Local/202@from-internal-eb35.2	SIP/202-08da16d0			NO ANSWER	0
2008-10-13 11:12:59	502		Zap/10-1	SIP/502-08da16d0			ANSWERED	39
2008-10-13 11:13:20	225	2512092	SIP/225-07d22420	Zap/7-1			ANSWERED	22
2008-10-13 11:14:00	502		Zap/10-1	SIP/502-08db940			ANSWERED	40

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 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.

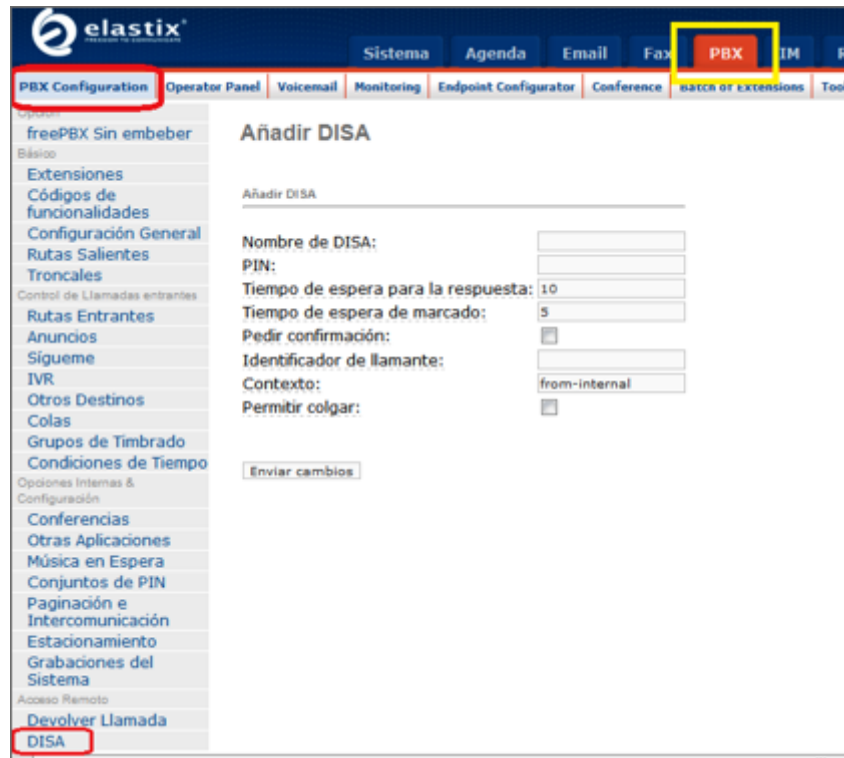


10.5 Configuración DISA

El módulo DISA (Direct Inward System Access) permite que los usuarios puedan acceder telefónicamente a los servicios provistos por la central Elastix desde el exterior, el acceso se realiza obteniendo 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 permitido).

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.



- Nombre de DISA: Nombre distintivo del acceso DISA. PIN: La clave para que el usuario acceda al soporte DISA (tono interno)
- Tiempo de espera para la respuesta: Tiempo máximo que la aplicación DISA espera para que el usuario complete el PIN
- Tiempo de espera de marcado: Máximo tiempo entre dígitos.
- Pedir Confirmación: Solicita confirmación antes de solicitar la contraseña.
- Identificador de Llamante: Si es utilizado este campo, todas las llamadas salientes originadas desde esta DISA utilizarán este número como CallerID.
- Contexto: Contexto que utilizarán las llamadas de la DISA.
- Permite Colgar: Si se habilita esta opción, el usuario que realiza una llamada por esa DISA tiene la posibilidad de colgar la llamada presionando "asterisco" para obtener un nuevo tono de marcado y poder realizar una nueva llamada.

Después, 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, día de la semana e inclusive el día del año. 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, es necesario saber que se debe implementar ingresando a dos menús. El primer menú llamado "Grupo Horario" se utiliza para definir

los distintos horarios que se necesitarán para discriminar las condiciones de tiempo, y luego el menú “Condiciones de Tiempo” que será utilizada para determinar el destino de la llamada de acuerdo al grupo horario anteriormente definido.

Para ingresar a la utilidad de “Grupo Horario”, debe dirigirse a la pestaña de “PBX → PBX Configuration → Grupo Horario”. Aquí accederemos a un formulario que nos permitirá establecer el rango de tiempo que será utilizado luego en el menú de “Condiciones de Tiempo”..



Después de definir el “Grupo Horario” es necesario ingresar a la utilidad de “Condiciones de Tiempo” para relacionar este grupo horario con una entidad como por ejemplo un IVR, una extensión, una cola, un voicemail, etc.

The screenshot shows the Elastix web interface. At the top, there are navigation tabs: Sistema, Agenda, Email, Fax, PBX (highlighted with a yellow box), and IM. Below these is a sub-menu: PBX Configuration (highlighted with a red box), Operator Panel, Voicemail, Monitoring, Endpoint Configurator, Conference, and Batch. On the left side, there is a sidebar menu with categories: Opción (freePBX Sin embeber), Básico (Extensiones, Códigos de funcionalidades, Configuración General, Rutas Salientes, Troncales), Control de Llamadas entrantes (Rutas Entrantes, Anuncios, Sigüeme, IVR, Otros Destinos, Colas), Grupos de Timbrado (Condiciones de Tiempo - highlighted with a red box), Configuración (Conferencias, Otras Aplicaciones, Música en Espera, Conjuntos de PIN, Paginación e Intercomunicación, Estacionamiento, Grabaciones del Sistema), and Acceso Remoto (Devolver Llamada, DISA). The main content area is titled 'Añadir condición horaria' and contains the following form fields:

- Añadir condición horaria: [Empty text field]
- Nombre de la condición horaria: Navidad
- Grupo horario: Navidad
- Day/Night Mode Association: [Empty text field]
- Associate with: No Association
- Destino si la hora coincide:
 - Phonebook Directory: Phonebook Directory
 - Queues: Sales <4000>
 - IVR: Unnamed
 - Terminate Call: Hangup
 - Extensions: <1001> Mickey Mouse
- Destino si la hora no coincide:
 - Phonebook Directory: Phonebook Directory
 - Queues: Sales <4000>
 - IVR: Unnamed
 - Terminate Call: Hangup
 - Extensions: <1001> Mickey Mouse
- Enviar button

Una vez configurada, no olvide seleccionarla dentro de sus rutas entrantes.

11 Funcionamiento interno de FreePBX

La libertad no se implora, se conquista

- Eloy Alfaro

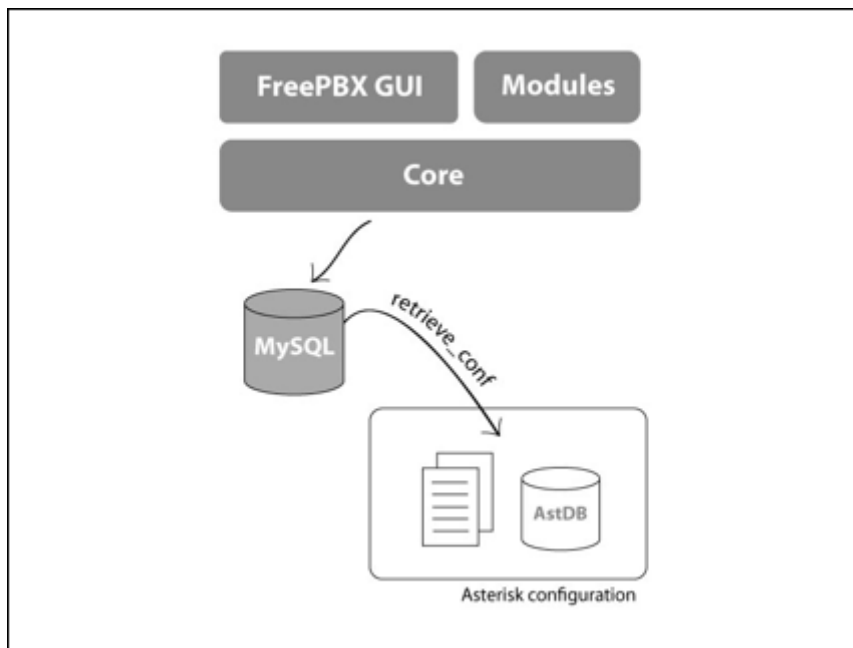
11.1 Fundamentos de FreePBX

FreePBX es un software de configuración de Asterisk que ha sido integrado como parte de la distro Elastix. Básicamente este software se encarga de generar los archivos de configuración requeridos por Asterisk a partir de la configuración ingresada por el usuario desde la interfaz Web.

Para realizar esta 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.
1. Convierte la información almacenada en MySQL en archivos de configuración vía la herramienta `retrieve_conf`

Para ilustrar este proceso veamos la siguiente figura.



11.2 Organización de archivos

FreePBX lee y escribe información en algunos directorios para su correcto funcionamiento. A continuación se encuentra una lista de los más importantes.

Directorio	Descripción
/var/www/html/admin/	Lugar donde reside el GUI de FreePBX
/etc/asterisk/	Archivos de configuración de Asterisk modificados por FreePBX
/var/lib/asterisk/agi-bin/	Lugar donde residen los <i>scripts</i> AGI de Asterisk. FreePBX cuenta con algunos <i>scripts</i> en esta carpeta como el <i>dialparties.agi</i>
/var/lib/asterisk/bin/	Algunos binarios o vínculos simbólicos correspondientes que son utilizados por FreePBX.
/usr/local/sbin/	El <i>script</i> amportal se encuentra aquí ya que solo puede ser manipulado por <code>root</code>
/var/lib/asterisk/sounds/	Sonidos que se utilizan en el plan de marcado por FreePBX
/var/lib/asterisk/sounds/custom/	Sonidos que han sido creados por el usuario. Por ejemplo cuando se crea un IVR de bienvenida
/var/spool/asterisk/voicemail/	Lugar donde se almacenan los <i>voicemails</i>
/var/spool/asterisk/monitor/	Lugar donde se almacenan las grabaciones de voz
/var/log/asterisk/cdr-csv/	Lugar donde se escriben los CDRs en formato texto

agagaa

11.3 Configuración

amportal.conf

El archivo **/etc/amportal.conf** es un archivo de texto plano que contiene información general de configuración y es leído por FreePBX. Es decir, este es el principal archivo de configuración de FreePBX y puede ser editado manualmente en caso de requerirse.

En este archivo de configuración se encuentra la información de acceso a la base de datos (usuario y contraseña) al igual que el tipo de autenticación que FreePBX usará para con sus usuarios Web. Este archivo está bien documentado por lo que no entraremos en detalles sobre 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 (MySQL en nuestro caso). Este programa

usualmente se ejecuta desde el Web a través de un script PHP cada vez que damos clic en la barra roja de FreePBX que dice Apply Configuration Changes Here. Sin embargo, retrieve_conf puede ser invocado inclusive desde la línea de comandos.

11.5 dialparties.agi

Este es un script AGI escrito en PHP que se invoca desde el plan de marcado cuando se quiere hacer una llamada. Su principal tarea es la de revisar el estado de la extensión a la que se está llamando, así como también ciertas opciones que pudieran estar configuradas en dicha extensión (call waiting, call forward, do not disturb). Finalmente, el script 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 rutear 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 (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

Esta sección tiene como propósito el que podamos entender mejor el funcionamiento de FreePBX con miras a solucionar posibles problemas.

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.<módulo>.php	Archivo principal del módulo conteniendo la funcionalidad necesaria. Como es de imaginar, la palabra <módulo> debe ser reemplazada por el respectivo nombre del módulo.
functions.inc.php	Librería de funciones PHP 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 Conectando Servidores Elastix

Cualquier poder, si no se basa en la unión, es débil

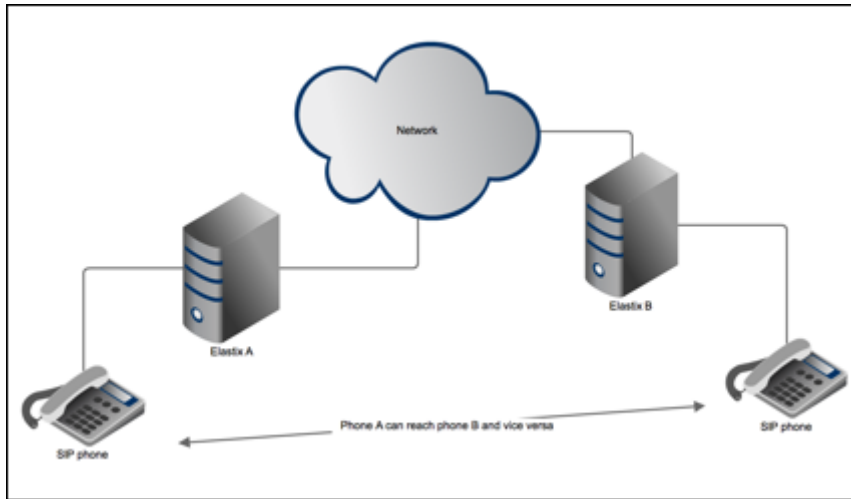
- Jean de la Fontaine

12.1 Interconexión entre dos servidores Elastix

Una de las ventajas de usar Elastix es que fácilmente podemos interconectar dos servidores Elastix de tal manera que lugares remotos se puedan comunicar con un solo plan de marcado.

A continuación veremos cómo interconectar dos servidores Elastix mediante una troncal de tipo SIP.

Supongamos que queremos interconectar un servidor Elastix localizado en un punto A con otro servidor Elastix en un punto B. El resultado debe ser que un teléfono conectado al servidor Elastix en el punto A pueda comunicarse marcando simplemente la extensión del teléfono conectado al servidor Elastix en el punto B, y viceversa.



Para implementar esto se requiere definir dos cosas:

1. La troncal SIP entre los servidores Elastix
2. Las rutas salientes en cada servidor Elastix

Toda la configuración se puede realizar desde la interfaz Web de Elastix desde el menú PBX → PBX Configuration

Definiendo la troncal SIP

Definimos el nombre de la troncal, el cual en nuestro caso servirá de username para el proceso de registro. Debemos crear la troncal del Elastix en el punto A como pointA y la del punto B como pointB. Ver capítulo 10 para detalles sobre la creación de troncales SIP.

Luego definimos los detalles del peer que en este caso serán los siguientes:

```
host=dynamic
secret=welcome
trunk=yes
type=friend
```

Esta configuración de la sección de peer se debe repetir en ambos servidores Elastix. En este ejemplo hemos definido como la palabra welcome como secret.

Ahora debemos definir el register string (cadena de registro) usando pointA o pointB como username según sea el caso.

Para el servidor Elastix en el punto A debemos tener:

```
pointB:welcome@192.168.1.240
```

Para el servidor Elastix en el punto B debería ser:

```
pointA:welcome@192.168.1.239
```

Una vez establecida esta configuración en ambos Elastix podemos verificar desde el Asterisk CLI si las troncales SIP se encuentran registradas. Al ejecutar el comando `sip show registry` en el servidor Elastix del punto A deberíamos ver algo como lo que sigue:

```
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 en la misma pantalla de creación de troncales SIP anterior. La sección de user establece donde se quiere terminar la llamada. En este caso queremos que sea accesible a extensiones locales, así que bastará con colocar el contexto primario `from-internal`. Esto indica que cuando la llamada entre por la troncal se iniciará en el contexto aquí especificado. La configuración es como sigue:

```
context=from-internal
host=192.168.1.240
insecure=very
type=friend
```

Definiendo las rutas salientes

Ver capítulo 10 para detalles sobre la creación de rutas salientes.

1. Primero determinamos el nombre de la ruta saliente. En este caso, especificaremos la ruta saliente en el servidor del punto A como: `ElastixB`
2. Después definimos el patrón de marcado. En este caso usaremos el siguiente patrón de marcado: `2|XXX`. Estamos asumiendo que para marcar a las extensiones conectadas al servidor Elastix en el punto B se debe marcar con el número 2 adelante.
3. Finalmente escogemos la troncal por la cual debe irse la llamada. En este caso escogemos la troncal `SIP/pointA`

12.2 Conexión con proveedores de llamadas vía IP

La popularidad de los proveedores de servicios de voz (Voice Service Providers o VSP) ha crecido increíblemente en los últimos años. Estas compañías proveen conexiones SIP o IAX para llamadas locales o internacionales por Internet a bajo costo.

En esta sección estableceremos una conexión SIP con un proveedor que nos permitirá hacer llamadas dentro de los Estados Unidos.

Nuestra meta es registrar nuestra troncal de tipo SIP con un VSP que nos permita realizar llamadas a USA desde cualquier teléfono IP conectado a nuestro servidor Elastix.

Para implementar esto se requiere definir dos cosas:

1. La troncal SIP del servidor Elastix
2. La ruta saliente en el servidor Elastix

Definiendo la troncal SIP

Como vimos anteriormente, lo primero que se debe hacer es definir el nombre de la troncal. En este caso le llamaremos: VP-SIP-NYC. Ver capítulo 10 para detalles sobre la creación de troncales SIP.

Ahora debemos definir las características del peer. Esta información debe ser proporcionada por el VSP. Generalmente, los VSP entregan esta misma información para todos sus usuarios con la única diferencia del username y el password.

Como ejemplo, nuestro VSP nos pidió que configuremos la troncal SIP 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

Ver capítulo 10 para detalles sobre la creación de rutas salientes.

1. Primero determinamos el nombre de la ruta saliente. En este caso la nombraremos: INTERNACIONAL
2. Luego definimos el patrón de marcado. En este caso usaremos el siguiente patrón de marcado: 00. Estamos asumiendo que para marcar un número al exterior se requiere anteponer los dos

ceros.

- 3. Finalmente escogemos la troncal por la cual debe irse la llamada. En este caso escogemos la troncal **SIP/VP-SIP-NYC**

Una vez establecida esta configuración podemos verificar desde el **Asterisk CLI** si la troncal SIP se encuentra registrada. Al ejecutar el comando **sip show registry** en el servidor Elastix deberíamos ver algo como lo que sigue:

```

elastix*CLI> sip show registry
Host                               Username           Refresh State     Reg.Time
nyc.voicepulse.com:5060            myusername        105 Registered   Fri,11 Apr 2008 10:29:43

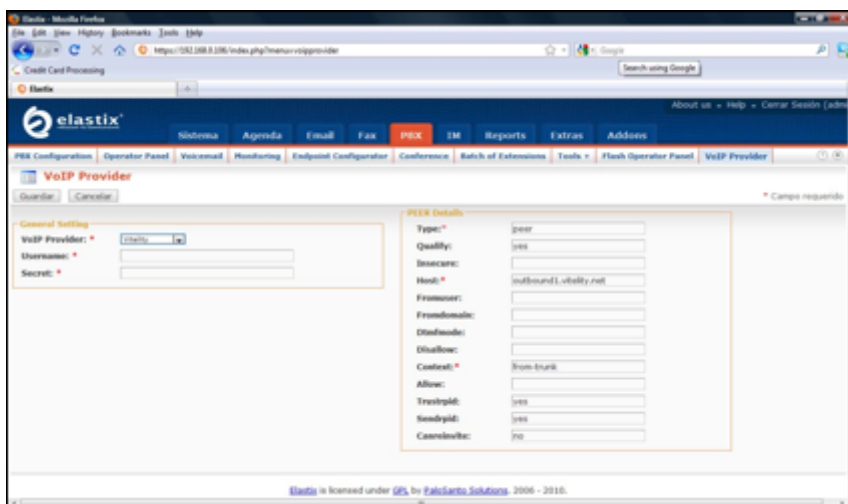
```

Al marcar un número internacional como 0018773527849 la llamada debe establecerse por medio de este VSP.

12.3 Conexión predefinida con proveedores de llamadas vía IP

Existe otra forma de crear troncales **SIP** para interconectar el servidor Elastix con un **VSP**. La interfaz de **VoIP Provider** nos permite ingresar los parámetros de peer y cadena de registro directamente en un formulario. Además, podemos seleccionar entre los proveedores más comunes los cuales ya cuentan con parámetros predeterminados de configuración.

La siguiente imagen muestra la configuración de una troncal **SIP** con un **VSP** predefinido.



13 Calidad de voz

Calidad es hacer las cosas bien cuando nadie está controlando

- Henry Ford

13.1 Síntomas de los problemas de voz

Enumerar los problemas que afectan la calidad de voz es difícil pues influye mucho la percepción de la persona. Las causas del deterioro de la calidad de voz son variadas y difíciles de categorizar. En esta sección revisaremos 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. El eco consiste en la repetición una o más veces la conversación con cierto desfase de tiempo.

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 4 hilos** (Ver capítulo 1 para más detalles). Esto sucede porque 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. Muchas de las tarjetas telefónicas disponibles para Asterisk no disponen de un buen mecanismo para el ajuste dinámico de la impedancia de la línea con la impedancia de la tarjeta. Por esta razón una parte de la onda se refleja causando eco.

- **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 programa llamado **fxotune** disponible en Elastix desde la línea de comandos. Lamentablemente esta utilidad se debe ejecutar manualmente y con el servicio Asterisk apagado. Para líneas con problemas de eco se recomienda ejecutarlo en horas no laborables una vez por semana debido a que puede llevarse un tiempo considerable. El comando es el siguiente.

```
fxotune -i 5
```

El comando **fxotune** prueba enviando una señal pura por la línea y escuchando el retorno. Esto lo hace muchas veces por cada línea hasta encontrar el mejor valor de ganancia. Al final, este software escribe el valor de la ganancia en el archivo **/etc/fxotune.conf**

Al finalizar su ejecución se debe instruir a **DAHDI** que cargue los valores nuevos de ganancia. Esto lo podemos hacer con el siguiente comando:

```
fxotune -s
```

Esto lo debemos ejecutar siempre antes de arrancar Asterisk por lo que se recomienda incluirlo al final del script de arranque de DAHDI ubicado en la ruta **/etc/init.d/dahdi** para no tener que hacerlo manualmente cada vez.

- **Nota:** Es muy importante hacer notar que si se utiliza **fxotune** para tratar de acoplar las líneas no se deben configurar las variables **txgain o rxgain** en el **chan_dahdi.conf** ya que

sobrescribirán el trabajo del fxtotune.

Ahora solo resta encender Asterisk nuevamente.

Finalmente, otra causa del eco es de naturaleza acústica 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 hace mucha diferencia y hay modelos que introducen menos eco acústico que otros.

Bajo nivel de volumen

Muchas redes telefónicas de baja calidad atenúan la señal de manera significativa haciendo que escuchemos 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 tonos de DTMFs.

Tanto el volumen de recepción como el de transmisión pueden ser amplificados mediante parámetros en el archivo `chan_dahdi.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 `fxtotune` para corregir el problema. Si modificamos los parámetros de ganancia arbitrariamente podríamos conseguir empeorar el eco en la línea.

El parámetro que modifica la ganancia de la señal de transmisión se llama **txgain** y el de recepción se llama **rxgain**. Ambos se pueden editar en el archivo **chan_dahdi.conf**.

Retardo de voz

El retardo de voz no es otra cosa que la demora que experimenta la voz en llegar a su destino. Usualmente el retardo es menor de un segundo y si es menor de 200 ms pasa desapercibido.

Cuando el retardo es mayor de 500 ms los interlocutores tienden a interrumpirse y la conversación se superponga. Esto es irritante para la comunicación..

Es casi imposible eliminar el retardo a nivel del servidor pues en la gran mayoría de los casos este es un síntoma de problemas inherentes en la red de comunicaciones. Es decir que si se quiere eliminarlo habrá que analizar posibles cambios en la red de comunicaciones.

Distorsión de voz

En esta categoría recaen diferentes problemas. El más común de ellos es el de la voz robotizada.

Cuando se utiliza un codec para optimizar el uso de ancho de banda (Ej: gsm) y la conectividad experimenta problemas de pérdida de información.

En estos casos es recomendable cambiar de codec a uno que ofrezca menor compresión de voz.

Comunicación entrecortada

Este problema tan molesto normalmente está relacionado con la pérdida de paquetes en la red. Las causas principales de la pérdida de paquetes son una **latencia elevada** o un **ancho de banda limitado**, entre otros. La comunicación entrecortada también puede ser ocasionada por un **jitter** elevado. Estudiaremos más sobre el jitter en la siguiente sección.

En todo caso lo normal es buscar la causa de este problema primero en la red y luego en el servidor. El entrecorte de la voz puede ser producto de problemas con el servidor cuando el consumo de los recursos como memoria y CPU se encuentran muy elevados.

Por lo general si la latencia máxima (absoluta, no promedio) de la red es 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 del retardo de voz sino del retardo de los paquetes en la red.

Una manera sencilla de calcular (aproximadamente) el retardo de la red es utilizando el comando ping. Este comando envía paquetes de prueba a un destino y nos presenta 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, promedio, máximo y su desviación.

Pérdida de paquetes

En algunas ocasiones ciertos de los paquetes que componen un mensaje no llegan a su destino. Pueden existir muchas razones para que suceda esto como equipos defectuosos o saturados, pérdidas en el medio de transmisión (cables mal terminados, ruido eléctrico elevado), entre otros.

En el resumen del comando ping en la sección de retardo de red vemos que también se nos reporta porcentualmente las pérdidas de paquetes. Lo deseable es que no existan pérdidas de paquetes en lo absoluto (0%). Si existen, hay que investigar el por qué.

Inclusive pérdidas de menos del 1% pueden afectar a las conversaciones de voz sobre IP. Este problema es aún mayor cuando usamos codecs de 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 de red y normalmente está en el orden de los milisegundos.

Este parámetro es necesario para analizar la calidad de voz ya que tener un promedio de retardo de red bajo no es suficiente para garantizar una buena calidad.

Cuando existe un promedio de retardo de red aceptable pero muy variable existe mayor probabilidad de que los paquetes lleguen en desorden a su destino o con retardos excesivos. Debido a que la comunicación de voz en tiempo real no puede esperar a que lleguen todos los paquetes y 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. Pese a que ping puede usarse en situaciones de apremio, el comando ping no envía paquetes de voz sino tráfico ICMP el cual no recibe el mismo tratamiento en la red. 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 minimizar los efectos negativos del jitter algunos fabricantes de soluciones de VoIP han pensado en esperar por paquetes de voz extraviados antes de volver a ensamblar la señal de voz en el destino. A medida que estos paquetes lleguen serán reordenados para reconstruir el mensaje original.

Esta solución no está libre de problemas ya que con esta acción estamos también aumentando el retardo de red al hacer que el mensaje llegue a su destino después del reordenamiento. Sin embargo la idea es buena y práctica. Quizás si esperamos sólo el tiempo 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 soporta el concepto de **jitter buffer** y este parámetro es configurable a nivel de protocolos IAX y SIP. Este parámetro se puede configurar 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.

En el **sip.conf**:

```
jbenable=yes  
jbuffer=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. La complejidad radica en que la calidad de voz es en parte un parámetro subjetivo de la persona que escucha. Incluso muchos conceptos como la comprensión de una conversación telefónica varían dependiendo del idioma que usemos. Adicionalmente, 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. 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. Esto ocurre porque nuestra mente le da más importancia a eventos más recientes que a los lejanos.

También hay personas que justifican inconscientemente cierta falta de calidad de voz en pro de las ventajas o conveniencias personales percibidas. Un ejemplo de esto es que las personas no se quejan mucho de la calidad de voz en teléfonos celulares 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 tiene mala calidad. Dicho de otra manera, como seres humanos inconscientemente estamos justificando esa disminución en la calidad de voz del teléfono celular.

Esto es solo la punta del iceberg pues para calificar la calidad de voz hay una serie de parámetros subjetivos un tanto difíciles de cuantificar. Lo importante de este ejemplo es ilustrar la difícil tarea que se tiene al tratar de medir la calidad de voz.

Aun así, existen algunos intentos de estandarizar la medición de la calidad de voz. Uno de los más conocidos es la escala **MOS** que se basa en mediciones subjetivas. También podemos mencionar el modelo **E** que ya pone en el tapete algunos parámetros objetivos como el retardo de red o la pérdida de paquetes. Algo interesante de este último modelo es que contempla la conversión de sus

resultados a la escala MOS lo que permite obtener una escala estándar 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**. Esta escala escribe una escala de calidad de voz basada en la toma de muestras subjetivas que se realizan con una serie de técnicas conocidas como **Absolute Category Rating (ACR)**.

Para tal efecto se reúne a un grupo de personas y se les pide que califiquen la calidad de voz de manera subjetiva. Antes de iniciar la evaluación se hacen escuchar algunos ejemplos que previamente han sido definidos por la recomendación de manera que los usuarios tengan algún marco referencial.

Una vez hecho esto se transmite una serie de frases (también pre-definidas por la recomendación) por la línea telefónica y los usuarios proceden a calificar la calidad de voz.

El siguiente es un resumen de 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

Una desventaja de usar la escala MOS es sin duda la cantidad de tiempo necesaria para determinar la calidad de voz de una simple línea. Imaginémoslo tratando de coordinar toda una serie de pruebas con un gran grupo de personas a quienes previamente tendremos que entrenar, y solo para evaluar la calidad de voz de una línea.

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 de red, pérdida de paquetes, y el jitter.

- **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. La calidad de voz contiene un importante componente subjetivo como lo hemos mencionado anteriormente. Sin embargo, muchas veces es más práctico medir los parámetros de red y hacer un cálculo matemático para aproximar la medida 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**. Esta recomendación dice que la calidad de voz queda representada por un parámetro **R** que se calcula de la siguiente manera:

$$R = R_0 - I_S - I_d - I_e + A$$

Donde,

- **R0** representa la relación señal-a-ruido (signal-to-noise ratio)
- **IS** representa la degradación que sufre la señal en su conversión a paquetes **Id** representa el retardo de red
- **le** representa la degradación introducida por los equipos de red y depende del codec y de las pérdidas de paquetes
- **A** representa el margen de seguridad

Fórmula simplificada

La recomendación ITU nos provee de una fórmula más simplificada para calcular el parámetro R. Se trata de una aproximación menos precisa que la expresión anterior 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.

Ya hemos definido los parámetros **Id** e **le** como el retardo de red y la pérdida de paquetes respectivamente. Sin embargo, antes de ingresar estos valores en la fórmula debemos normalizar estos valores de acuerdo a la recomendación ITU.

El parámetro **Id** representa el retardo de red y se normaliza con la siguiente fórmula.

$$I_d = 0.024 \cdot d + 0.11 \cdot (d - 177.3) \cdot H(d - 177.3)$$

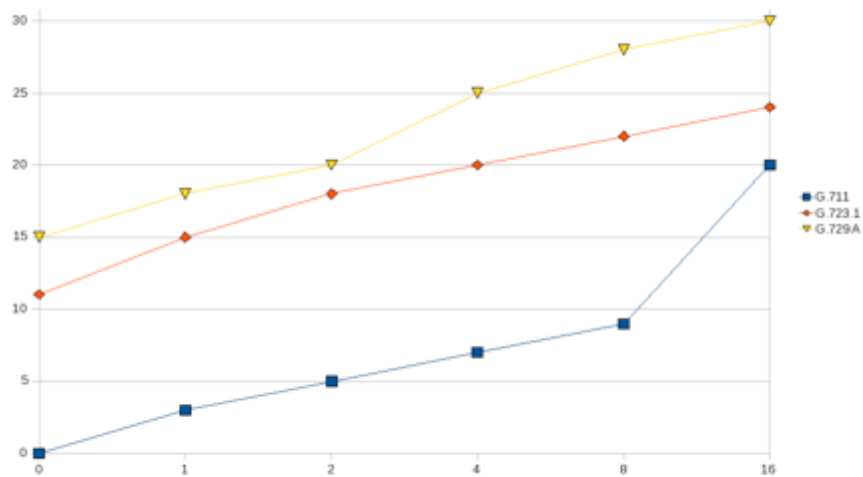
Donde **d** es el retardo de red en sí expresado en milisegundos y **H()** es la función de **Heavyside**:

$$H(x) = 0 \text{ para } x < 0 \text{ y } 1 \text{ para } x \geq 0.$$

Al analizar la fórmula de **Id** notaremos a primera vista que hay un **valor de inflexión en los 177.3 ms**. Esto quiere decir que 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 de siempre mantener el retardo de red por debajo de 170 ms (aproximadamente).

Finalmente debemos calcular **le** para poder reemplazar este valor en la fórmula y **calcular R**.

le depende principalmente de la pérdida de paquetes y del codec usado. Para mantener las cosas relativamente simples usaremos un gráfico con el cual podremos determinar el valor aproximado de **le** dependiendo de los dos factores mencionados. La tabla es la siguiente:



Ie (y) vs. pérdida de paquetes (x)

Este gráfico fue generado a partir de la siguiente tabla de valores aproximados. El eje vertical representa el parámetro **Ie** mientras que el eje horizontal representa la pérdida de paquetes.

Pérdida de paquetes %	Ie [G.711]	Ie [G.723.1]	Ie [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 de paquetes ya algunos codecs merman la calidad de voz.

Traduciendo R a la escala MOS

Habiendo calculado el valor de R procedemos a explicar cómo traducirlo a la más conocida escala MOS.

La fórmula es la siguiente:

MOS = 1 ; R < 0

MOS = 1 + 0.035 · R + 7 · R · (R – 60) · (100 – R) · 10⁻⁶ ; 0 < R < 100

MOS = 4.5 ; R > 100

Ahora tenemos una manera práctica y objetiva de medir la calidad de voz en una instalación de voz sobre IP. De esta manera podremos hacer los ajustes necesarios para mejorar la satisfacción de los usuarios telefónicos.

- **Ejemplo**

Supongamos que hemos realizado mediciones en un punto de la red y queremos determinar la calidad de voz en dicho punto conociendo que usamos el codec **G.711**. El **retardo** de red es de **143 ms** y la **pérdida de paquetes** llega al **7%**.

¿Cuál es la medida de calidad de voz según la escala MOS?

Después de manipular los valores de acuerdo a las expresiones expuestas anteriormente, llegaremos a un valor similar al siguiente.

$$R = 94.2 - 0.024 \cdot 143 - 7.2$$

$$R = 83.57$$

Traduciendo R a la escala MOS obtenemos:

$$MOS = 4.15$$

Lo cual es un resultado aceptable (4.15/5.00) pues todavía encontraremos usuarios satisfechos.

13.4 Canceladores de eco

Funcionamiento de canceladores de eco

Es muy importante saber cómo funcionan los canceladores de eco para entender cómo sacarles provecho al máximo y quizás para diseñar el nuestro propio o modificar alguno existente en caso de ser código abierto.

Existen implementaciones de canceladores de eco tanto en hardware como en software. Generalmente se prefiere que la cancelación de eco sea implementada en hardware para no sobrecargar el procesador del servidor. Los principios de cancelación de eco rigen de igual forma para cualquiera de las dos implementaciones.

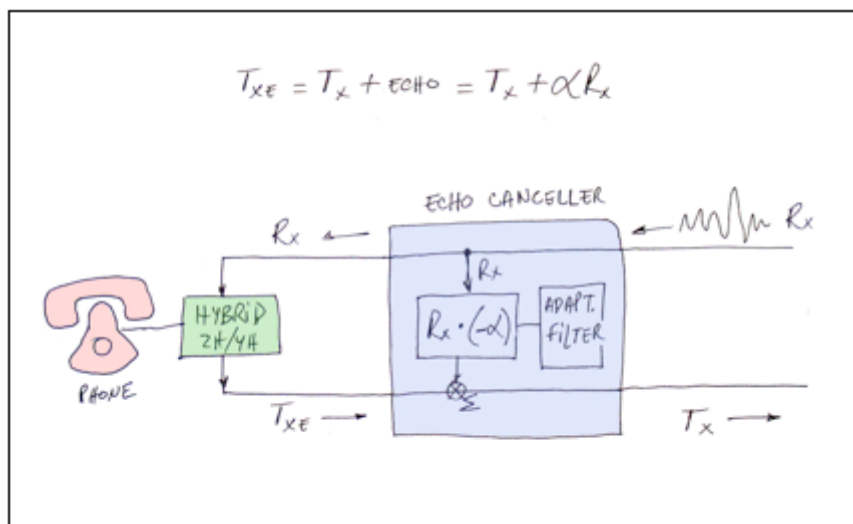
- Un **cancelador** de eco parte de un principio lógico muy sencillo: Si el eco significa que una parte de la señal de ida se refleja en la de regreso, entonces **para eliminar el eco** debería bastar con **restar la señal de ida** (en cierta proporción) **de la señal de regreso**.

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). Se llama así porque 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 R_x , la señal transmitida es T_x y la señal transmitida con el componente de eco es T_{xe} .



Funcionamiento básico 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. Por lo tanto, también tenemos que predecir este retardo para poder sustraer la señal de eco. Predecir el retardo no es una tarea sencilla y por lo general los canceladores de eco tienen limitantes. Revisar la señal en busca de retardo por grandes lapsos de tiempo conlleva demasiado trabajo y los canceladores de eco pueden dejar de funcionar.

- **Nota:** Hay que hacer un pequeño paréntesis para recordar que los canceladores de eco deben funcionar en tiempo real y por lo tanto no pueden perder el tiempo en cálculos complejos ya que dejarían de ser veloces y retrasarían el envío de la voz.

Los valores típicos de retardo que un cancelador de eco puede manejar son 64 ms o 128 ms. Obviamente que mientras menor sea el retardo, 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. Se podría 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 es fácil determinar si un cancelador de eco es compatible o no. Inclusive existen empresas que se dedican al

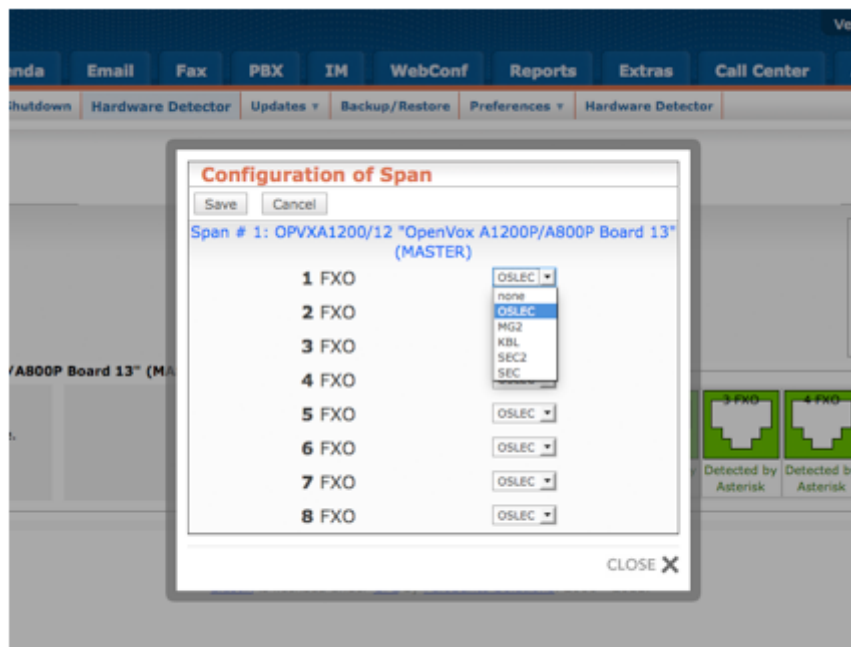
tedioso trabajo de revisar si un cancelador de eco es compatible o no con G.168.

No vamos a explicar aquí los detalles de G.168 pues sale del alcance del presente libro. Para mayor información sobre detalles técnicos de G.168 recomendamos visitar el sitio del ITU (<http://www.itu.int>).

OSLEC

Elastix incorpora un interesante cancelador de eco por software de libre distribución conocido como **OSLEC**. Este cancelador de eco es parte de DAHDI y se puede configurar independientemente para cada canal de manera sencilla desde la interfaz Web de Elastix.

Esta es una imagen de la interfaz de detección de hardware desde donde podemos activar el soporte para cancelación de eco.



Configuración de cancelador de eco en interfaz de Elastix

14 Protocolos VoIP al descubierto

Cualquier persona, en mar o en tierra, con un aparato sencillo y barato que cabe en un bolsillo, podría recibir noticias de cualquier parte del mundo o mensajes particulares destinados sólo al portador, la Tierra se asemejaría, pues, a un inconmensurable cerebro, capaz de emitir una respuesta desde cualquier punto.

- Nikola Tesla, 1905

14.1 Protocolo SIP

Session Initiation Protocol (SIP) es un protocolo de señalización (application-layer control) creado para administrar sesiones multimedia entre dos o más partes. Contrario a lo que se pueda pensar, este protocolo no es quien transporta la voz propiamente dicha sino que se encarga de la comunicación necesaria para establecer, modificar y finalizar una llamada.

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".

SIP es un protocolo **peer-to-peer**. Es decir que prácticamente toda la lógica es almacenada en los puntos de terminación de la llamada o **endpoints**.

Detalles del protocolo

Usa a su vez dos importantes protocolos:

- **RTP** transmite la voz y el video
- **SDP** negociar las capacidades de los endpoints

La última versión del estándar es la **RFC3261**. Permite entender fácilmente los mensajes transmitidos al enviarlos como texto, lo que facilita la depuración (debugging).

Señalización SIP

El protocolo SIP utiliza métodos y respuestas para cumplir con su función. Un método es el mecanismo empleado para convenir una acción. La respuesta, como su nombre lo indica, contiene la respuesta a un método.

Métodos

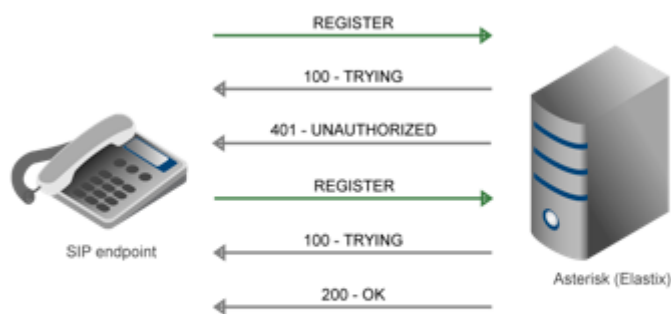
Método	Descripción
INVITE	Invita a un usuario a una llamada
ACK	Abreviación de <i>acknowledgement</i> (acuse de recibo) enviada para indicar que se ha recibido un mensaje
BYE	Termina una conexión entre usuarios o rechaza una llamada
CANCEL	Termina el requerimiento o búsqueda de un usuario
OPTIONS	Solicita información acerca de las capacidades del servidor SIP
REGISTER	Registra la ubicación de un usuario
INFO	Intercambia información de señalización en el transcurso de la sesión

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. 401 Unauthorized, 404 Not Found, 482 Loop Detected)
- 5xx Server Failure (e.g. 501 Not Implemented)
- 6xx Global Failure (e.g. 603 Decline)

Registración

El siguiente diagrama ilustra la **registración** de un **endpoint SIP** contra un servidor SIP. Debe tenerse en cuenta que esta es una vista simplificada ya que en el proceso de registración también se debe negociar el **códec** que será usado además de otros parámetros.



Registración SIP

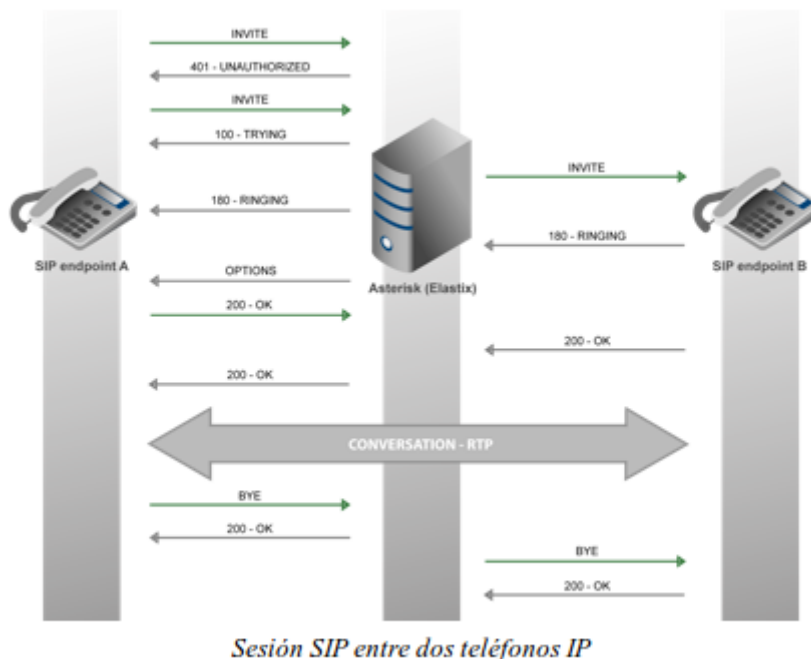
Esta registración debe llevarse a cabo periódicamente de tal forma que durante la sesión no se tenga

que enviar información de autenticación.

Sesión

Luego de registrarse, el **endpoint SIP** puede iniciar una sesión (en nuestro caso una llamada telefónica).

En la siguiente imagen ilustraremos el caso específico de una conversación entre dos endpoints SIP. Se han omitido intencionalmente mensajes como **ACKs** para mejorar la legibilidad.



Hay que notar que **SIP** se encarga de la señalización solamente. La voz propiamente dicha se transmite por el protocolo **RTP**, el cual explicaremos con mayor detalle al final de este capítulo.

Como podemos observar toda la comunicación recala en el servidor Elastix. Es decir, Elastix hace las veces de mediador (**SIP Proxy**).

SIP y Asterisk

Algunos detalles que debemos tener en cuenta cuando se usa el protocolo SIP con Asterisk son los siguientes:

- Versiones anteriores de Asterisk (hasta 1.4 inclusive) limitaban el funcionamiento del protocolo SIP a UDP. En versiones actuales Asterisk ya podemos hacer uso de SIP inclusive sobre TCP.
- Por omisión se usa el **puerto 5060** para el servicio **SIP** aunque este parámetro se puede modificar en el archivo **sip.conf**
- **SIP** adolece de problemas de **Network Address Translation (NAT)** aunque se puede combatir este problema con parámetros como **externip**, **localnet** y **nat** en el archivo **sip_nat.conf**
- En Asterisk es posible hacer diagnóstico del protocolo SIP.

14.2 Protocolo IAX

El **Inter-Asterisk eXchange (IAX)** es un protocolo de señalización creado por **Mark Spencer**, el mismo creador de Asterisk. Su objetivo es el de solucionar los problemas existentes con otros protocolos. Este protocolo todavía no es un estándar en la industria pero pretende serlo a través de un proceso de estandarización en la IETF.

En esencia **IAX presenta dos ventajas muy interesantes** sobre otras alternativas de protocolo como **SIP**.

- Consume menos ancho de banda
- **Soluciona** mejor los problemas de **NAT** y **firewalls**

Revisemos cada una de estas ventajas.

Consume menos ancho de banda

Mientras SIP es un protocolo basado en texto, **IAX es un protocolo binario**. Esto es una ventaja desde el punto de vista del consumo de ancho de banda puesto que en binario se desperdiciarán menos bytes. Además, el protocolo **IAX 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. Esto significa una reutilización de datagramas y por consiguiente un ahorro adicional de ancho de banda.

Soluciona mejor los problemas de NAT y firewalls

SIP transmite señalización por el puerto 5060 y usa pares de puertos entre el 10000 y 20000 para la voz en sí. Esta separación de puertos tiende a causar problemas al pasar la comunicación de voz por un firewall. En cambio **IAX transmite en UDP** y usa solamente el **puerto 4569** para transmitir tanto la señalización como la voz. Esto permite conducir comunicaciones a través de firewalls con mucha más facilidad. El administrador debe asegurarse de abrir un solo puerto en el firewall. Además, el endpoint IAX no necesita saber nada sobre la red en la cual está operando. Esta combinación de atributos hace que se elimine el caso donde la llamada se completa pero el audio no pasa.

Si reflexionamos acerca de estas ventajas nos daremos cuenta rápidamente que resultan perfectas para troncalización entre dos servidores Elastix. En otras palabras, cuando se trata de **interconectar** dos o más **servidores Elastix** entre sí el uso de **IAX es altamente recomendable**..

La versión actual del protocolo IAX es la 2. La anterior ha quedado obsoleta por lo que es común ver el nombre IAX2 como sinónimo de IAX.

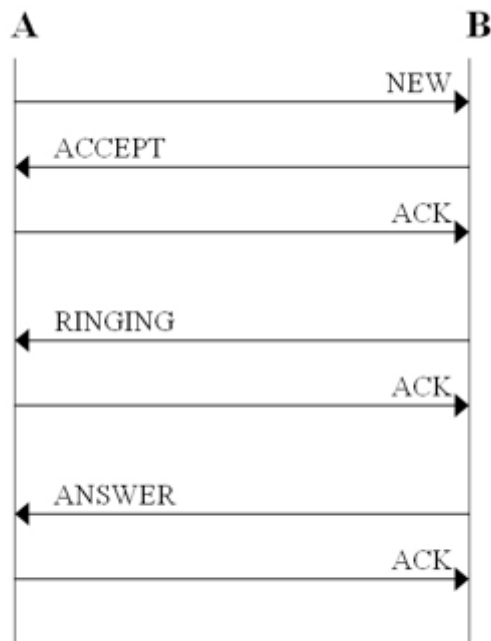
Fases de una llamada IAX

Una llamada IAX tiene tres fases que se explican a continuación.

Establecimiento de la llamada

Para iniciar una llamada el equipo que la inicia (equipo A) le envía un **mensaje NEW** al equipo que la recibe (equipo B). Éste último responde con un **mensaje ACCEPT**. Después de esto el equipo **B** timbrará su campana esperando que el usuario levante el auricular. 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 esta explicación para añadir claridad a la explicación.



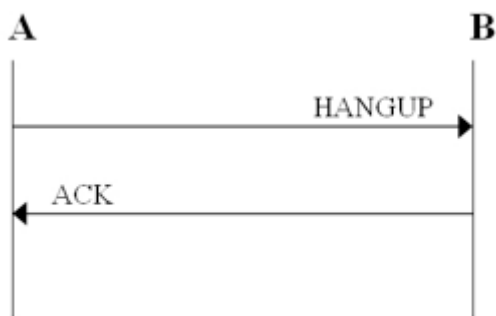
Establecimiento de una llamada IAX

Llamada en curso

Una vez contestada la llamada por el equipo **B** se inicia el intercambio de audio mediante unos paquetes llamados **frames**. Estos frames se envían dentro del mismo flujo de comunicación que la señalización inicial.

Colgado

Cualquiera de las partes involucradas en la conversación puede enviar un mensaje **HANGUP** para terminar la llamada en cualquier momento.

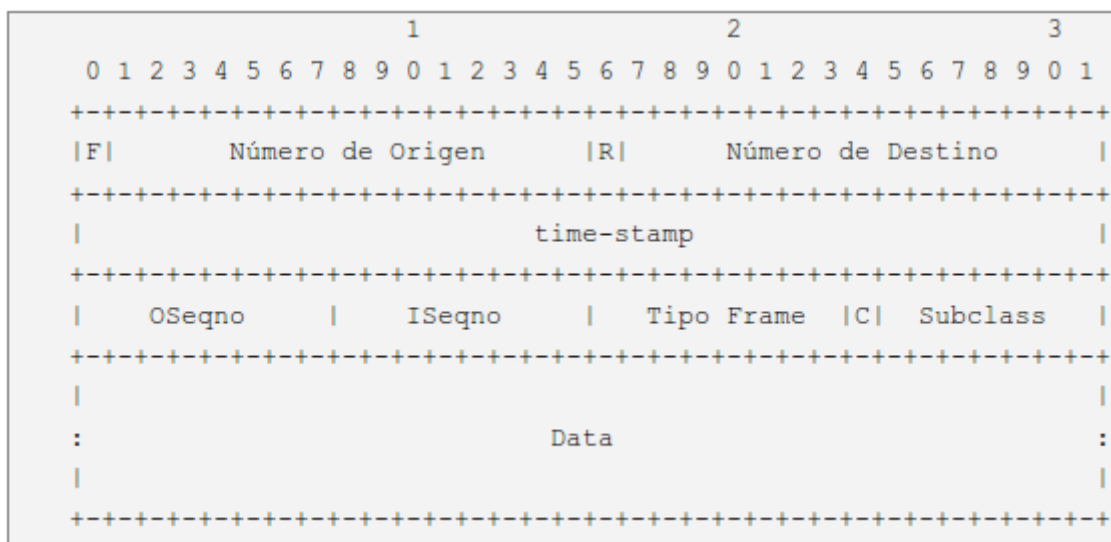


Colgado de una llamada IAX

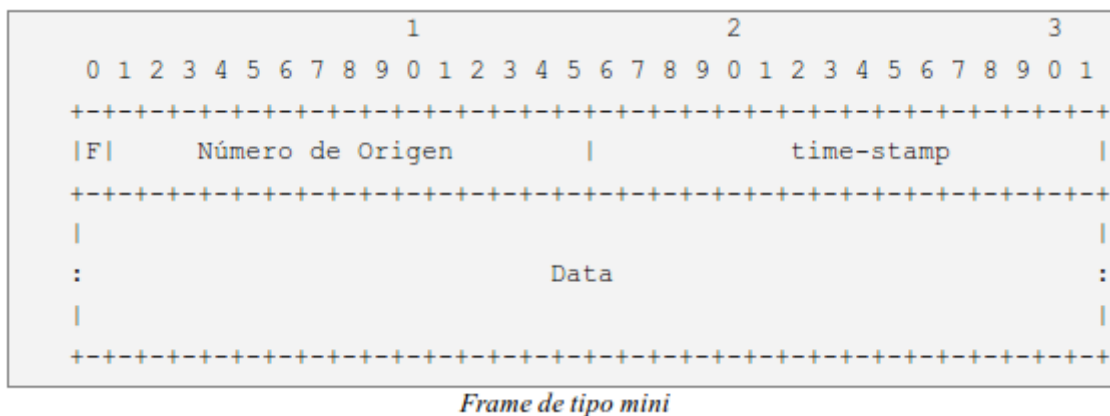
Frames

Para intercambiar el audio entre los participantes de una llamada se pueden utilizar dos tipos de frames específicos para IAX llamados **full** y **mini**. También se les suelen llamar **F** y **M** por sus iniciales.

Una conversación está compuesta en su mayoría por **frames de tipo mini**, cuya virtud es (como su nombre lo sugiere) ser pequeños y ligeros. Esto quiere decir que tienen una **cabecera** pequeña de **4 bytes**, lo cual ayuda a ahorrar ancho de banda. De cuando en cuando se intercambian **frames tipo full**, los cuales conllevan adicionalmente **información de sincronización** para mantener sincronizados a ambos puntos.



Frame de tipo full



14.3 Protocolo RTP

RTP es el protocolo que se encarga de **transportar la voz** propiamente dicha. Muchas personas se confunden y piensan que ese es el trabajo de SIP pero no es así. Como ya lo hemos visto anteriormente en este capítulo, una vez que SIP establece una llamada es RTP quien toma la iniciativa para transportar la voz a su destino.

RTP trabaja sobre UDP y por lo tanto no hay mucho control de transmisión. Es decir que el equipo emisor envía la voz hacia el otro extremo con solo la esperanza de que llegue. Por tanto no espera recibir confirmación del mensaje recibido ya que la voz necesita ser transmitida en tiempo real y cualquier demora puede comprometer la calidad de voz. Si un paquete de **voz se 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**.

RTP cuenta con la **ayuda** de otro protocolo llamado **RTCP**. Este protocolo no es del todo indispensable pero proporciona valiosa ayuda al momento de transportar la voz de manera óptima ya que proporciona estadísticas e información de control. Esta información a su vez le permite a los equipos de voz involucrados en la conversación tomar decisiones para mejorar la transmisión en caso de ser posible. Para este efecto los paquetes RTCP se transmiten periódicamente.

Estructura de un paquete RTP

Un paquete RTP se compone de un **encabezado** y la data o **payload**. El siguiente diagrama ilustra la estructura del encabezado de los paquetes RTP.

Byte 1				Byte 2		Byte 3	Byte 4
V=2	P	X	CC	M	PT	Número de Secuencia	
Timestamp							
Fuente de Sincronización (SSRC)							
Fuente Contribuyente (CSRC)							
Extensión (Opcional dependiendo del bit X)							
Data							

Estos campos se definen de la siguiente forma.

- V es el número de versión. Este campo es de 2 bits de longitud y su valor siempre es el número 2.
- P es un bit que indica si hay relleno o padding al final de la data. El relleno no es otra cosa que bytes adicionales al final del payload utilizados para ciertos algoritmos de encriptación que esperan un payload de tamaño fijo.
- 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 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 comunicándose con diferentes fuentes de paquetes RTP. Es un número aleatorio de 32 bits por lo que hay la posibilidad (aunque sea 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.
- Extensión es un campo de longitud variable que permite extender la funcionalidad de la cabecera. Esta información solo se adjunta a la cabecera RTP si el bit X lo indica.

Después de la cabecera se deben transmitir los datos que no son otra cosa que el audio propiamente dicho convertido a digital y codificado.

Cabe recordar que **RTP** utiliza **dos puertos para la comunicación**, uno de ida y otro para el retorno. Este par de puertos es negociado mientras se establece la llamada entre los puntos pero en el caso de Asterisk está dentro del rango 10000 a 20000. Sin embargo, esto es configurable en el archivo `/etc/asterisk/rtp.conf`.

14.4 Protocolo SRTP

SRTP o RTP seguro (**Secure Real-time Protocol**) es, como su nombre lo indica, un alternativa a RTP cuando necesitamos cifrar la información transmitida. En nuestro caso se vuelve muy útil cuando se

necesita garantizar que una llamada no podrá ser escuchada por un interceptor de la misma, manteniendo la información confidencial.

Desde el punto de vista de un protocolo de tiempo real se hace un poco difícil concebir un cifrado de datos que no añada retardo a la comunicación. Si cada paquete debe ser **encriptado** en el origen y **des-encriptado** en destino, sin duda esto debe añadir un **retardo adicional**, dependiendo de la potencia de nuestro CPU que es quien realizará la matemática necesaria para el cifrado de datos. Además también podemos pensar que el uso de algoritmos de cifrado añadirá ancho de banda adicional, lo cual siempre es indeseable.

Todos los temores anteriores son válidos y también estuvieron en la mente de quienes diseñaron el protocolo. Es por eso que si leemos el documento oficial donde se define este protocolo (RFC 3711) veremos que aparte de los objetivos obvios de SRTP (confidencialidad de información e integridad de datos) se definen estos otros:

- Bajo impacto en el ancho de banda
- Bajo costo computacional

En general podemos decir que hicieron un buen trabajo y el protocolo no añade un costo computacional o ancho de banda significativo por lo que en la gran mayoría de escenarios podremos implementar SRTP sin ningún temor.

SRTP está disponible de manera **nativa en Asterisk** recién desde la versión 1.8 y habilitarlo es bastante sencillo. Por ejemplo, si la comunicación es SIP basta añadir a la configuración de cada **endpoint** el parámetro **encryption=yes**

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.

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 tanto el nombre de los archivos de log como el nivel de detalle de la información que se escribirá en ellos. Esto se hace en una sola línea con la **siguiente sintaxis**.

```
Nombre de archivo => Niveles de debugging
```

Donde los niveles de **debugging** son separados por coma y 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**". Este nombre representa la consola de Asterisk, o sea lo que se muestra en el Asterisk CLI.

El siguiente es 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
; The DTMF log is very handy if you have issues with IVR's
;dtmf => dtmf
;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** activado 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**

Acorde al archivo **logger.conf**, Elastix habilita por defecto el 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_DE_ARCHIVO: MENSAJE

A continuación veremos un ejemplo del archivo de log de Asterisk.

```

root@elastix:~# tail -f /var/log/asterisk/full
[Oct 29 22:47:17] VERBOSE[27887] logger.c: -- Executing [s@ivr-2:1] Set("Zap/5-1", "LOOPCOUNT=0") in new stack
[Oct 29 22:47:17] VERBOSE[27887] logger.c: -- Executing [s@ivr-2:2] Set("Zap/5-1", "DIR_CONTEXT=default") in new stack
[Oct 29 22:47:17] VERBOSE[27887] logger.c: -- Executing [s@ivr-2:3] Set("Zap/5-1", "IVR_CONTEXT_ivr-2=0") in new stack
[Oct 29 22:47:17] VERBOSE[27887] logger.c: -- Executing [s@ivr-2:4] Set("Zap/5-1", "IVR_CONTEXT_ivr-2") in new stack
[Oct 29 22:47:17] VERBOSE[27887] logger.c: -- Executing [s@ivr-2:5] GotoIf("Zap/5-1", "0!begin") in new stack
[Oct 29 22:47:17] VERBOSE[27887] logger.c: -- Executing [s@ivr-2:6] Answer("Zap/5-1", "") in new stack
[Oct 29 22:47:17] DEBUG[27887] chan_zap.c: Took Zap/5-1 off hook
[Oct 29 22:47:17] DEBUG[27887] chan_zap.c: Engaged echo training on channel 5
[Oct 29 22:47:17] VERBOSE[27887] logger.c: -- Executing [s@ivr-2:7] Wait("Zap/5-1", "1") in new stack
[Oct 29 22:47:17] VERBOSE[27887] logger.c: == Spawn extension (ivr-2, s, 7) exited non-zero on 'Zap/5-1'
[Oct 29 22:47:17] VERBOSE[27887] logger.c: -- Executing [s@ivr-2:3] Hangup("Zap/5-1", "") in new stack
[Oct 29 22:47:17] VERBOSE[27887] logger.c: == Spawn extension (ivr-2, h, 1) exited non-zero on 'Zap/5-1'
[Oct 29 22:47:17] VERBOSE[27887] logger.c: -- Hangup 'Zap/5-1'
[Oct 29 22:47:28] NOTICE[4869] chan_sip.c: -- Registration for 'pFW3tkg68@fk-primary.voicepulse.com' timed out, trying again (attempt # 6824)
[Oct 29 22:47:28] NOTICE[4854] chan_sax2.c: Failed to authenticate on REGISTER to 'pFW3tkg68@fk-primary.voicepulse.com' (Tries 1)
[Oct 29 22:47:28] NOTICE[4854] chan_sax2.c: Restricting registration for peer '518' to 60 seconds (requested 300)
[Oct 29 22:47:28] NOTICE[4854] chan_sax2.c: Restricting registration for peer '8222' to 60 seconds (requested 300)
[Oct 29 22:47:28] DEBUG[4846] dsp.c: dsp busy pattern set to 8,8
[Oct 29 22:47:28] VERBOSE[27889] logger.c: -- Starting single switch on 'Zap/5-1'
[Oct 29 22:47:28] NOTICE[27889] chan_zap.c: Got event 17 (Polarity Reversal)....
[Oct 29 22:47:27] NOTICE[27889] chan_zap.c: Got event 17 (Polarity Reversal)....
[Oct 29 22:47:27] NOTICE[27889] chan_zap.c: Got event 17 (Polarity Reversal)....
[Oct 29 22:47:27] VERBOSE[27889] logger.c: -- Executing [s@from-zaptel:1] NoOp("Zap/5-1", "Entering from-zaptel with DID == ") in new stack
[Oct 29 22:47:27] VERBOSE[27889] logger.c: -- Executing [s@from-zaptel:2] Ringing("Zap/5-1", "") in new stack
[Oct 29 22:47:27] VERBOSE[27889] logger.c: -- Executing [s@from-zaptel:3] Set("Zap/5-1", "DID=8") in new stack
[Oct 29 22:47:27] VERBOSE[27889] logger.c: -- Executing [s@from-zaptel:4] NoOp("Zap/5-1", "DID is now 8") in new stack
[Oct 29 22:47:27] VERBOSE[27889] logger.c: -- Executing [s@from-zaptel:5] GotoIf("Zap/5-1", "1!zapok!notzap") in new stack
[Oct 29 22:47:27] VERBOSE[27889] logger.c: -- Goto (from-zaptel,s,8)
[Oct 29 22:47:27] VERBOSE[27889] logger.c: -- Executing [s@from-zaptel:8] NoOp("Zap/5-1", "Is a ZapTel channel") in new stack
[Oct 29 22:47:27] VERBOSE[27889] logger.c: -- Executing [s@from-zaptel:9] Set("Zap/5-1", "CHANV-1") in new stack
[Oct 29 22:47:27] VERBOSE[27889] logger.c: -- Executing [s@from-zaptel:10] Set("Zap/5-1", "CHANV-1") in new stack
[Oct 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
[Oct 29 22:47:27] WARNING[27889] app_macro.c: No such context 'macro-from-zaptel-5' for macro 'from-zaptel-5'

```

Monitoreo del log

Es muy importante dar seguimiento en tiempo real a la actividad del servidor Elastix. Muchas veces nos veremos haciendo esto mientras realizamos o recibimos una llamada para tratar de averiguar qué ha fallado.

El siguiente es un método útil para revisar interactivamente el archivo de **log full** a medida que se va escribiendo con mensajes de Asterisk.

```
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 las advertencias podemos modificar ligeramente el filtro de la siguiente manera.

```
tail -f /var/log/asterisk/full | grep -E "(ERROR|WARNING)"
```

- **Nota:** Si queremos examinar todo el log en lugar de hacerlo interactivamente debemos reemplazar el comando tail -f por el comando cat

¿Qué información nos puede indicar el campo MENSAJE?

El **MENSAJE** se refiere a la información que viene a la derecha de cada línea del log luego del nombre del archivo. Si se trata de una línea de ERROR o WARNING encontraremos aquí una breve descripción del error ocurrido o de la advertencia. Sin embargo existen otras 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.

Estas líneas de nivel VERBOSE (además de otras líneas informativas) nos proveen de información muy útil para diagnosticar problemas.

Especialmente útiles son las que empiezan con la palabra Executing ya que a continuación mencionan el contexto y la prioridad en el que Asterisk está procesando el plan de marcado. La sintaxis de estas líneas es la siguiente.

```
[TIME DATE] VERBOSE[PID] logger.c: -- Executing  
[extension@contexto:prioridad] MAS_TEXTO
```

MAS_TEXTO puede contener información acerca de la aplicación que fue ejecutada junto 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). Además, la extensión era la **s** y la prioridad era **7**. También podemos observar que se está tratando de marcar mediante la aplicación Dial() usando ciertos parámetros. En fin, podemos observar que hay mucha información útil que podemos obtener de cada línea escrita en el log.

Número de proceso (PID)

Asterisk es un software multiproceso. Esto quiere decir que puede crear varias instancias de sí mismo que trabajen de manera coordinada entre sí para ejecutar diferentes tareas al mismo tiempo. Cuando Asterisk escribe una línea en el log también anota el **número de proceso (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 el PID si es que lo conocemos.

```
cat /var/log/asterisk/full | grep 38484
```

Con el comando anterior estamos filtrando la salida para mostrar el proceso 38484. Por supuesto que también podremos haber capturado alguna cadena de caracteres que coincida fortuitamente con el patrón 38484. Sin embargo estas líneas son irrelevantes y se las puede eliminar manualmente.

Gracias a este procedimiento podremos filtrar el hilo de una llamada y aislarlo para encontrar errores. Esto es particularmente útil en un ambiente con alto número de llamadas simultáneas debido a la gran cantidad de información que Asterisk escribe en el log.

- **Nota:** Algo importante a tener en cuenta es que una llamada puede tener más de un proceso (y por consiguiente PID) involucrado.

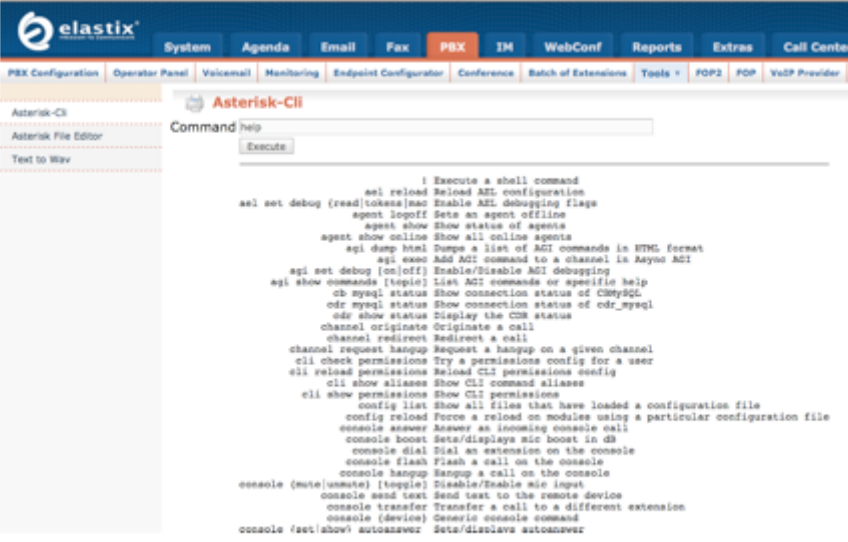
15.2 Diagnóstico a través del CLI

Desde el Asterisk CLI podemos **ver en tiempo real el funcionamiento de Asterisk**. Es algo similar a hacer un "tail -f /var/log/asterisk/full" pero con la ventaja de poder ver el texto de salida resaltado (para mejor legibilidad). Además el CLI nos permite interactuar simultáneamente con Asterisk al poder ejecutar algunos comandos desde la consola.

Elastix nos brinda una interfaz directa al **Asterisk CLI ubicada en PBX→Tools** y basada en Web. Usar el CLI desde la interfaz Web puede resultar muy cómodo al no tener que iniciar una sesión SSH, la cual muchas veces se encuentra restringida para conexiones remotas. Lo que hay que tener en cuenta es que desde la interfaz Web no es posible tener un CLI interactivo. Esta es la principal diferencia de ejecutar el CLI desde la línea de comandos a ejecutarlo desde Web.

Para acceder al CLI desde la línea de comandos es necesario ejecutar lo siguiente.

```
asterisk -r
```



```

| Execute a shell command
aef reload Reload AEL configuration
aef set debug (read/toksa) Enable AEL debugging flags
agent logoff Sets an agent offline
agent show Show status of agents
agent show online Show all online agents
agi dump.html Dumps a list of AGI commands in HTML format
agi exec Add AGI command to a channel in Anyus AGI
agi set debug (on/off) Enable/Disable AGI debugging
agi show commands [topic] List AGI commands or specific help
cdr mysql status Show connection status of CDRmysql
cdr mysql status Show connection status of cdr_mysql
channel originate Originate a call
channel redirect Redirect a call
channel request hangup Request a hangup on a given channel
cli check permissions Try a permissions config for a user
cli reload permissions Reload CLI permissions config
cli show aliases Show CLI command aliases
cli show permissions Show CLI permissions
config list Show all files that have loaded a configuration file
config reload Force a reload on modules using a particular configuration file
console answer Answer an incoming console call
console boost Sets/displays mic boost in dB
console dial Dial an extension on the console
console flash Flash a call on the console
console hangup Hangup a call on the console
console (mute/unmute) [toggle] Disable/Enable mic input
console send text Send text to the remote device
console transfer Transfer a call to a different extension
console (device) Generic console command
console /set/show/ autoanswer Sets/displays autoanswer

```

Ejecutando el CLI a través de la interfaz Web de Elastix

15.3 Sniffing

El término **sniffing** describe la forma de 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 soluciones de código abierto que hacen un gran trabajo como es el caso de **Wireshark**.

Wireshark era anteriormente conocido como Ethereal. Este poderoso sniffer contiene herramientas de análisis avanzadas que nos permitirán **analizar paquetes RTP y SIP** entre otras cosas.

Para capturar los detalles de una llamada y analizarla debemos primero definir la tarjeta de red del servidor en la cual queremos escuchar si existe más de una. La captura de paquetes en sí es muy sencilla y solo debemos ejecutar Wireshark de acuerdo a sus instrucciones.

Finalizada la captura lo más probable es que junto con los paquetes de voz hayamos capturado otros paquetes que circulaban por la red. Muchos de estos paquetes como ICMP, DNS, y SMTP no son de nuestro interés pero con algo de práctica aprenderemos a filtrar estos paquetes irrelevantes.

16 Control externo de Asterisk

Más que exigirlo, el respeto es algo que uno se gana

- Anónimo

Muchas veces editar el plan de marcado no es suficiente y hace falta algún mecanismo más flexible para lograr lo que la empresa necesita. En otras ocasiones quisiéramos inclusive poder controlar Asterisk en tiempo real a través de alguna aplicación remota.

Para expandir el comportamiento de Asterisk más allá del plan de marcado existen dos **interfaces** muy útiles llamadas **AGI** y **AMI**., Estas interfaces nos permiten interactuar con Asterisk a un nivel mucho más poderoso que el logrado hasta ahora.

16.1 Asterisk AGI

Asterisk Gateway Interface (AGI) establece una manera de interactuar con Asterisk desde un programa de línea de comandos. Este programa puede ser escrito en prácticamente cualquier lenguaje de programación y es invocado por Asterisk desde el plan de marcado o 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 gozan de mucha popularidad. Elastix viene por defecto con algunos de estos **scripts preinstalados** en la carpeta **/var/lib/asterisk/agi-bin/**

Ejecución de scripts AGI

Los scripts AGI se comunican con Asterisk a través de los descriptores de archivos **STDIN, STDOUT y STDERR**. Esto quiere decir que deberían poderse ejecutar desde la misma línea de comandos.

Algo muy importante de mencionar es que los **scripts AGI** deben tener **permisos de ejecución** para el usuario asterisk y el grupo del mismo nombre para que puedan correr. Podemos asignar los permisos de la siguiente forma.

```
# chown asterisk.asterisk script_prueba.agi
# chmod 755 script_prueba.agi
```

Después 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 teclee la extensión 6789. Por omisión Asterisk asume que el script se encuentra en la ruta **/var/lib/asterisk/agi-bin/** aunque se puede especificar otra ruta en la función AGI().

Parámetros enviados al script AGI

Como habíamos dicho antes, Asterisk le envía cierta información al script AGI al momento de su invocación. A continuación un ejemplo de la información enviada al script **dialparties.agi** que es un componente de **FreePBX** cuando lo ejecutamos.

```
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_callingtms: 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 Asterisk le envía al script AGI por omisión, nosotros podríamos enviarle información adicional invocando la función AGI() con ciertos parámetros de la siguiente manera.

```
exten => 6789,1,Answer
exten => 6789,2,AGI(script_prueba.agi,param1,param2,param3)
```

En el ejemplo anterior se le envían los parámetros adicionales **param1**, **param2** y **param3** al script. El script puede recibir tantos parámetros como sean necesarios.

Comandos AGI

Para la referencia del lector, a continuación mostramos un listado de los comandos AGI disponibles en la última versión de Elastix. Este listado ha sido generado con el comando **agi show commands** desde el CLI.

Comando	Descripción
answer	Contesta el canal
asynccgi break	Interrumpe el Async AGI
channel status	Retorna el estado del canal conectado
database del	Elimina clave/valor de la base de datos
database deltree	Elimina clave de árbol/valor de la base de datos
database get	Obtiene un valor de la base de datos
database put	Añade/actualiza un valor de la base de datos
exec	Ejecuta una aplicación
get data	Escucha por tonos DTMF en un canal
get full variable	Evalúa una expresión de canal
get option	Stream de archivo, escucha por DTMF, con timeout
get variable	Obtiene una variable de canal
hangup	Cuelga el canal actual
noop	No hace nada
receive char	Recibe un caracter de los canales que lo soporten
receive text	Recibe texto de los canales que lo soporten
record file	Graba a un archivo
say alpha	Vocaliza una cadena de caracteres
say digits	Vocaliza una cadena de dígitos
say number	Vocaliza números
say phonetic	Vocaliza una cadena de caracteres fonéticamente
say date	Vocaliza una fecha
say time	Vocaliza una hora
say datetime	Vocaliza una hora en un formato específico
send image	Envía imágenes a canales que lo soporten

<code>send text</code>	Envía texto a canales que lo soporten
<code>set autohangup</code>	Auto-cuelga el canal después de un tiempo
<code>set callerid</code>	Especifica el callerid para el canal actual
<code>set context</code>	Especifica el contexto para el canal
<code>set extension</code>	Cambia la extensión del canal
<code>set music</code>	Habilita/Deshabilita el generador de música de espera
<code>set priority</code>	Especifica la prioridad en el plan de marcado para el canal
<code>set variable</code>	Especifica una variable de canal
<code>stream file</code>	Envía un archivo de audio en el canal
<code>control stream file</code>	Envía un archivo de audio en el canal y permite al receptor controlar el stream
<code>tdd mode</code>	Cambia el modo de TDD (para sordos)
<code>verbose</code>	Escribe en mensaje en el archivo de <i>log verbose</i> de Asterisk
<code>wait for digit</code>	Espera el ingreso de un dígito
<code>speech create</code>	Crea un objeto de reconocimiento de voz (speech)
<code>speech set</code>	Especifica una opción del motor de reconocimiento de voz
<code>speech destroy</code>	Destruye un objeto de reconocimiento de voz (speech)
<code>speech load grammar</code>	Carga un vocabulario
<code>speech unload grammar</code>	Descarga un vocabulario
<code>speech activate grammar</code>	Activa un vocabulario
<code>speech deactivate grammar</code>	Desactiva un vocabulario
<code>speech recognize</code>	Reconoce la voz
<code>gosub</code>	Ejecuta una subrutina del plan de marcado

Librerías AGI

A pesar de que un script AGI puede ser escrito prácticamente en cualquier lenguaje, existen algunos lenguajes que ya cuentan con librerías para facilitar las labores de desarrollo. Algunos de estos lenguajes son los siguientes.

- Perl
- PHP
- Python Librerías AGI
- Ruby
- C
- .NET

Para mayor información sobre las librerías o **frameworks** disponibles para estos lenguajes recomendamos revisar el siguiente vínculo: <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 en la siguiente sección.

Ejemplo sencillo de script AGI

Para ilustrar mejor el funcionamiento de un script AGI, codificaremos algo muy básico. Este script vocaliza un número cuando marcamos una extensión determinada.

Veamos a continuación el código en PHP.

```
#!/usr/bin/php -q
<?php
    echo "SAY NUMBER 1234 \n";
    sleep(4);
?>
```

Este código lo podemos guardar como `simple.agi` en la ruta estándar `/var/lib/asterisk/agi-bin/`

Una vez guardado debemos estar seguros que el script pueda ser ejecutado por el usuario `asterisk` del grupo `asterisk`. Como lo hemos explicado anteriormente, esto se consigue de la siguiente manera.

```
chown asterisk.asterisk /var/lib/asterisk/agi-bin/simple.agi
chmod 755 /var/lib/asterisk/agi-bin/simple.agi
```

Finalmente debemos modificar el plan de marcado para invocar nuestro script AGI cuando se marque una extensión determinada. Usaremos la extensión `6789` para este ejemplo.

```
exten => 6789,1,Answer
exten => 6789,2,AGI(simple.agi)
exten => 6789,3,Hangup
```

Estas instrucciones podrían ubicarse al final del archivo `/etc/asterisk/extensions_custom.conf`, ya que este archivo no es sobrescrito por FreePBX.

- ¡**Listo!** Marque la extensión `6789` y escuchará el código decir **“one two three four”**.

16.2 Asterisk AMI

El **Asterisk Manager Interface (AMI)** establece una manera de comunicarse con Asterisk a través del **protocolo IP**. Digamos que es un concepto similar al AGI pero orientado para su ejecución desde

equipos remotos. Es esto último lo que lo convierte en una herramienta poderosa.

Con la funcionalidad que provee AMI se pueden desarrollar poderosas aplicaciones cliente que pueden ejecutarse en otros computadores liberando de carga al servidor Asterisk. Un ejemplo de lo que se puede hacer con AMI 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 **AMI permite que equipos remotos controlen el funcionamiento de Asterisk**, se requiere que estos equipos se autentiquen. Los usuarios, claves y permisos se definen en un archivo llamado **/etc/asterisk/manager.conf**

A continuación reproducimos parte del archivo **manager.conf** que se distribuye con Elastix por defecto.

```
;
; AMI - Asterisk Manager interface
;
[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,config,command,dtmf,reporting,cdr,dialplan,originate
write = system,call,log,verbose,command,agent,user,config,command,dtmf,reporting,cdr,dialplan,originate

#include manager_additional.conf
#include manager_custom.conf
```

En este archivo se define un usuario llamado **admin** con clave **elastix456** que puede conectarse desde la dirección IP 127.0.0.1 con máscara 255.255.255.0. Luego de esto se definen ciertos permisos para este usuario.

Probando desde telnet

Para probar la interacción con AMI no es necesario de un programa sofisticado. Una manera sencilla de hacerlo es a través del programa **telnet** disponible en cualquier computador.

Veamos un ejemplo ejecutando este comando 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/1.1
```

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 1.1 del protocolo.

A continuación nos autenticamos con el usuario y clave especificado en el archivo **manager.conf**. Básicamente debemos escribir lo siguiente en el terminal.

```
Action: Login
ActionID: 1
Username: admin
Secret: elastix456
```

Es importante conocer que después de escribir lo anterior y para enviar el comando debemos ingresar dos retornos de carro. Es decir que **debemos presionar Enter dos veces** para enviar la petición de autenticación anterior. Luego de esto AMI nos contestará con un mensaje de éxito o error. El siguiente es el mensaje que obtenemos como exitoso.

```
Response: Success
ActionID: 1
Message: Authentication accepted
```

De ahora en adelante comenzarán a aparecer mensajes de registro que el AMI envía a nuestra sesión cada vez que ocurre un evento. Por ejemplo, recibimos estos mensajes cuando una llamada se termina o si una extensión está timbrando.

Una vez autenticada la sesión podemos ejecutar comandos, también llamados acciones, directamente vía AMI. Tipo de paquetes AMI

En la comunicación por AMI se pueden identificar tres tipos de paquetes de información.

1. Acción: Un comando enviado por el cliente al servicio AMI para que sea procesado.
2. Respuesta: La respuesta de Asterisk a la acción requerida por el cliente.
3. Evento: La información de un evento asíncrono ocurrido en Asterisk. Por ejemplo, la llegada de 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 obtiene al ejecutar el comando **manager show commands**

desde el CLI de Elastix.

Comando	Privilegios	Descripción
AbsoluteTimeout	system,call,all	Especifica el <i>timeout</i> absoluto
AgentLogoff	agent,all	Especifica que un usuario ha dejado de estar autenticado
Agents	agent,all	Enlista los agentes y sus estados
AGI	agi,all	Añade un comando AGI para ejecución con Async AGI
Atxfer	call,all	Transferencia atendida
Bridge	call,all	Conecta dos canales presentes en el PBX
Challenge	<none>	Genera reto (<i>challenge</i>) para autenticación de MD5
ChangeMonitor	call,all	Cambia el archivo de monitoreo de un canal
Command	command,all	Ejecuta un comando del CLI
CoreSettings	system,reporting,all	Muestra configuración del core del PBX (versión, etc)

CoreShowChannels	system,reporting,all	Enlista los canales activos actualmente
CoreStatus	system,reporting,all	Muestra las variables de estado del <i>core</i> del PBX
CreateConfig	config,all	Crea un archivo vacío en el directorio de configuración
DAHDI DialOffhook	<none>	Marcar sobre un canal DAHDI mientras está en <i>offhook</i>
DAHDI DNDoff	<none>	Cambia el estado de no molestar (DND) del canal DAHDI a OFF
DAHDI DNDon	<none>	Cambia el estado de no molestar (DND) del canal DAHDI a ON
DAHDI Hangup	<none>	Termina el canal DAHDI
DAHDI Restart	<none>	Reinicia completamente los canales DAHDI (termina todas las llamadas)
DAHDI ShowChannels	<none>	Muestra el estado de los canales DAHDI

DAHDITransfer	<none>	Transfiere un canal DAHDI
DBDel	system,all	Elimina un registro de la base de datos
DBDelTree	system,all	Elimina un árbol de la base de datos
DBGet	system,reporting,all	Obtiene un registro de la base de datos
DBPut	system,all	Ingresa un registro en la base de datos
Events	<none>	Controla el flujo de un evento
ExtensionState	call,reporting,all	Verifica el estado de una extensión
GetConfig	system,config,all	Obtiene la configuración
GetConfigJSON	system,config,all	Obtiene la configuración en formato JSON
Getvar	call,reporting,all	Obtiene una variable de canal
Hangup	system,call,all	Termina el canal
IAXnetstats	system,reporting,all	Muestra las estadísticas de red de IAX
IAXpeerlist	system,reporting,all	Enlista los <i>peers</i> de IAX
IAXpeers	system,reporting,all	Enlista los <i>peers</i> de IAX

IAXregistry	system,reporting,all	Muestra la registraciones de IAX
ListCategories	config,all	Enlista las categorías del archivo de configuración
ListCommands	<none>	Enlista los comandos de manager (AMI) disponibles
Login	<none>	Inicia la sesión del Manager
Logoff	<none>	Termina la sesión del Manager
MailboxCount	call,reporting,all	Revisa el número de mensajes en el buzón
MailboxStatus	call,reporting,all	Revisa el buzón
MeetmeList	reporting,all	Enlista los participantes en una conferencia
MeetmeMute	call,all	Calla un usuario de <i>Meetme</i>
MeetmeUnmute	call,all	Permite hablar a un usuario de <i>Meetme</i>
ModuleCheck	system,all	Verifica si un módulo está cargado
ModuleLoad	system,all	Carga un módulo

Monitor	call,all	Monitorea un canal
Originate	originate,all	Origina una llamada
Park	call,all	Parquea un canal
ParkedCalls	<none>	Enlista las llamadas parqueadas
PauseMonitor	call,all	Pausa el monitoreo de un canal
Ping	<none>	Comando para mantener activa la conexión
PlayDTMF	call,all	Reproduce señal DTMF en un canal específico
QueueAdd	agent,all	Añade una interface a la cola
QueueLog	agent,all	Añade un record personalizado en el queue_log
QueuePause	agent,all	Establece un miembro de una cola como temporalmente no disponible
QueuePenalty	agent,all	Establece la penalidad para un miembro de una cola

QueueReload	<none>	Recarga una cola, colas, o cualquier sub-sección de una cola o colas
QueueRemove	agent,all	Elimina una interface de la cola
QueueReset	<none>	Vacía las estadísticas de la cola
QueueRule	<none>	Reglas de la cola
Queues	<none>	Colas
QueueStatus	<none>	Estado de la cola
QueueSummary	<none>	Resumen de la cola
Redirect	call,all	Redirige (transfiere) una llamada
Reload	system,config,all	Envía un evento de recarga
SendText	call,all	Envía un mensaje de texto al canal
Setvar	call,all	Establece una variable de canal
ShowDialPlan	config,reporting,all	Enlista el plan de marcado
SIPnotify	system,all	Envía una notificación de SIP
SIPpeers	system,reporting,all	Enlista los <i>peers</i> de SIP (formato texto)

<code>SIPqualifypeer</code>	system,reporting,all	Muestra los <i>peers</i> de SIP (formato texto)
<code>SIPshowpeer</code>	system,reporting,all	Muestra los <i>peers</i> de SIP (formato texto)
<code>SIPshowregistry</code>	system,reporting,all	Muestra las registraciones SIP (formato texto)
<code>Status</code>	system,call,reporting,all	Enlista el estado del canal
<code>StopMonitor</code>	call,all	Termina el monitoreo de un canal
<code>UnpauseMonitor</code>	call,all	Libra de pausa el monitoreo de un canal
<code>UpdateConfig</code>	config,all	Actualiza la configuración básica
<code>UserEvent</code>	user,all	Envía un evento arbitrario
<code>VoicemailUsersList</code>	call,reporting,all	Enlista toda la información de usuarios de <i>voicemail</i>
<code>WaitEvent</code>	<none>	Espera a que un evento ocurra

Anexo: 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 (Interfaz 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.
adtranvoifr.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 <code>AlarmReceiver()</code> .
alsa.conf	Configura el <i>driver</i> de sonido con los <i>drivers</i> de ALSA.
amd.conf	Se utiliza para detección automática de máquinas contestadoras.
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 CBMySQL (<i>Conference Bridge MySQL</i>) usado para manejar las conferencias en Elastix.
cdr.conf	El archivo principal de configuración de la reportación de CDRs.
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>).

<code>cdr_mysql.conf</code>	Archivo de configuración del módulo <code>cdr_mysql</code> que permite registrar los CDRs en la base de datos MySQL.
<code>cdr_odbc.conf</code>	Aquí se configura el módulo que permite registrar CDRs a través del <i>driver</i> ODBC de base de datos.
<code>cdr_pgsql.conf</code>	Aquí se configura el módulo que permite registrar CDRs en la base de datos PostgreSQL.
<code>cdr_tds.conf</code>	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.
<code>chan_dahdi.conf</code>	Este archivo contiene la configuración de los canales tipo DAHDI. Este archivo también incluye los archivos <code>chan_dahdi_additional.conf</code> y <code>dahdi-channels.conf</code> .
<code>chan_dahdi_additional.conf</code>	Este archivo lo sobrescribe FreePBX principalmente cuando se crean extensiones tipo DAHDI.
<code>cli.conf</code>	Archivo de configuración de la consola de Asterisk.
<code>codecs.conf</code>	En este archivo se configuran algunas opciones que controlan el funcionamiento de algunos <i>codecs</i> .
<code>dahdi-channels.conf</code>	Este archivo se encuentra incluido en el <code>chan_dahdi.conf</code> y se trata de un archivo generado automáticamente por el Hardware Detector de Elastix a través de la herramienta <code>dahdi_genconf</code> .
<code>dnsmgr.conf</code>	Este archivo define si Asterisk debe realizar peticiones DNS regularmente y cada cuánto tiempo.
<code>dundi.conf</code>	Archivo de configuración de DUNDI (<i>Distributed Universal Number Discovery</i>).
<code>enum.conf</code>	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.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 <code>extensions_additional.conf</code> y a <code>extensions_custom.conf</code> .
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.
features.conf	Aquí se pueden configurar algunas características de Asterisk como parámetros de transferencias y captura de llamadas.
features_applicationmap_additional.conf	Archivo incluido desde el <code>features.conf</code> que contiene configuración de la sección <code>applicationmap</code> y que es sobrescrito por FreePBX.
features_applicationmap_custom.conf	Archivo incluido desde el <code>features.conf</code> que contiene configuración de la sección <code>applicationmap</code> y que puede ser sobrescrito por el usuario ya que no lo sobrescribe FreePBX.
features_featuremap_additional.conf	Archivo incluido desde el <code>features.conf</code> que contiene configuración de la sección <code>featuremap</code> y que es sobrescrito por FreePBX.
features_featuremap_custom.conf	Archivo incluido desde el <code>features.conf</code> que contiene configuración de la sección <code>featuremap</code> 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.
festival.conf	Archivo de configuración de <i>Festival</i> .
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 <i>Google Talk</i> .
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.conf	Archivo principal de IAX que incluye a los otros.
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.

<code>iax_general_custom.conf</code>	Similar al <code>iax_general_additional.conf</code> pero aquí puede escribir información el usuario ya que este archivo no es sobrescrito por FreePBX.
<code>iax_registrations_custom.conf</code>	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 sobrescriba.
<code>iax_registrations.conf</code>	Archivo utilizado por FreePBX para almacenar información de cadenas de registro con otras centrales o proveedores VoIP que usan protocolo IAX.
<code>iaxprov.conf</code>	Archivo utilizado par actualizar el <i>firmware</i> de dispositivos IAX.
<code>indications.conf</code>	Archivo para configurar la definición de tonos. Estos tonos pueden variar de un país a otro o de una compañía telefónica a otra.
<code>jabber.conf</code>	Archivo que permite configurar el soporte XMPP en Asterisk. Esto es porque Asterisk se puede conectar como cliente a un servidor XMPP (El nombre Jabber es el anterior nombre del protocolo XMPP).
<code>jingle.conf</code>	Archivo de configuración de Jingle. Jingle es una extensión del protocolo XMPP que permite soportar multimedia a través de este protocolo.
<code>localprefixes.conf</code>	Archivo que permite configurar reglas de marcado basadas en prefijos.
<code>logger.conf</code>	En este archivo se puede configurar el <i>logging</i> de Asterisk. Ver la sección de diagnóstico para mayor información.
<code>manager.conf</code>	Archivo de configuración del servicio de <i>Asterisk Manager Interface</i> .
<code>manager_additional.conf</code>	Información de configuración de AMI que puede ser sobrescrita por FreePBX.
<code>manager_custom.conf</code>	Información de configuración de AMI para uso del usuario, o sea que este archivo no es sobrescrito por FreePBX.

meetme.conf	Configuración de los cuartos de conferencia permanentes.
meetme_additional.conf	Configuración de cuartos de conferencia escrita por FreePBX.
mgcp.conf	Configuración de protocolo MGCP.
misdn.conf	Archivo de configuración del canal chan_misdn en Asterisk.
modem.conf	Archivo de configuración del soporte para <i>modems</i> 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.conf	Aquí se configura la música en espera. Asterisk es muy flexible e inclusive soporta <i>streaming</i> de audio.
musiconhold_additional.conf	Archivo 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.
muted.conf	Permite enviar audio a ciertos canales y reducir el volumen de dicho audio mientras estos canales están activos.
osp.conf	Permite 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	Permite configurar canales tipo OSS (<i>Open Sound System</i>). OSS es un <i>driver</i> de audio (o un conjunto de <i>drivers</i> de audio) y nos permite configurar canales en la propia de tarjeta de sonido. Algo similar al soporte ALSA.
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 <code>phone</code> 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 <code>PrivacyManager()</code> .
queues.conf	Archivo general de configuración para colas. Si desea editar las colas en archivos de texto, se recomienda modificar <code>queues_custom.conf</code> .
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.
res_mysql.conf	Permite a Asterisk obtener la configuración de la base de datos MySQL. El concepto se denomina <i>Asterisk Realtime</i> .
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.conf	Archivo de configuración de SIP que incluye a los otros.
sip_additional.conf	Configuración de SIP generada por FreePBX y que no se debe editar manualmente.
sip_custom.conf	Configuración de SIP que se puede modificar y no es sobre escrita por FreePBX.
sip_general_additional.conf	Archivo complementario al <code>sip.conf</code> 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.
sip_general_custom.conf	Similar al <code>sip_general_additional.conf</code> pero aquí puede escribir información el usuario ya que este archivo no es sobrescrito por FreePBX.
sip_nat.conf	Configuración para trabajar con SIP a través de NAT (<i>Network Address Translation</i>).
sip_notify.conf	Utilizado para configurar el soporte que permite reiniciar algunos teléfonos SIP remotamente.
sip_registrations_custom.conf	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.
sip_registrations.conf	Archivo utilizado por FreePBX para almacenar información de cadenas de registro con otras centrales o proveedores VoIP.
skinny.conf	Permite configurar el protocolo SCCP (<i>Skinny Client Control Protocol</i>) en Asterisk.
sla.conf	Configuración de SLAs (<i>Shared Line Appearances</i>).

smdi.conf	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.
udptl.conf	Le permite a Asterisk soportar paquetes UDPTL. Estos paquetes son utilizados por el protocolo T.38 para fax.
unicall.conf	Aquí se puede configurar el módulo <code>chan_unicall</code> .
users.conf	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 mismo usuario puede cambiar de dispositivo telefónico y desde cualquier dispositivo se puede autenticar.
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.

El siguiente grupo de archivos puede ser encontrado en versiones de Elastix menores a 2.0, cuando todavía se utilizaban los drivers de Zapitel.

Archivo	Descripción
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 Zapitel.

- Editado por **Juan Carlos Ballesteros**, profesor de ciclo formativo de grado superior STI, IES Palomeras-Vallecas, en Marzo de 2014
- Fuente :[Comunicaciones Unificadas con Elastix - Volumen 1 - Segunda Edición](#)

From: <http://server-jk.ddns.net/dokuwiki/> - **IES Palomeras-Vallecas Dep. Electronica**

Permanent link: http://server-jk.ddns.net/dokuwiki/doku.php?id=sistemas_de_telefonia_fija_y_movil:teoria:telefonía_con_elastix

Last update: **2025/01/22 02:02**

